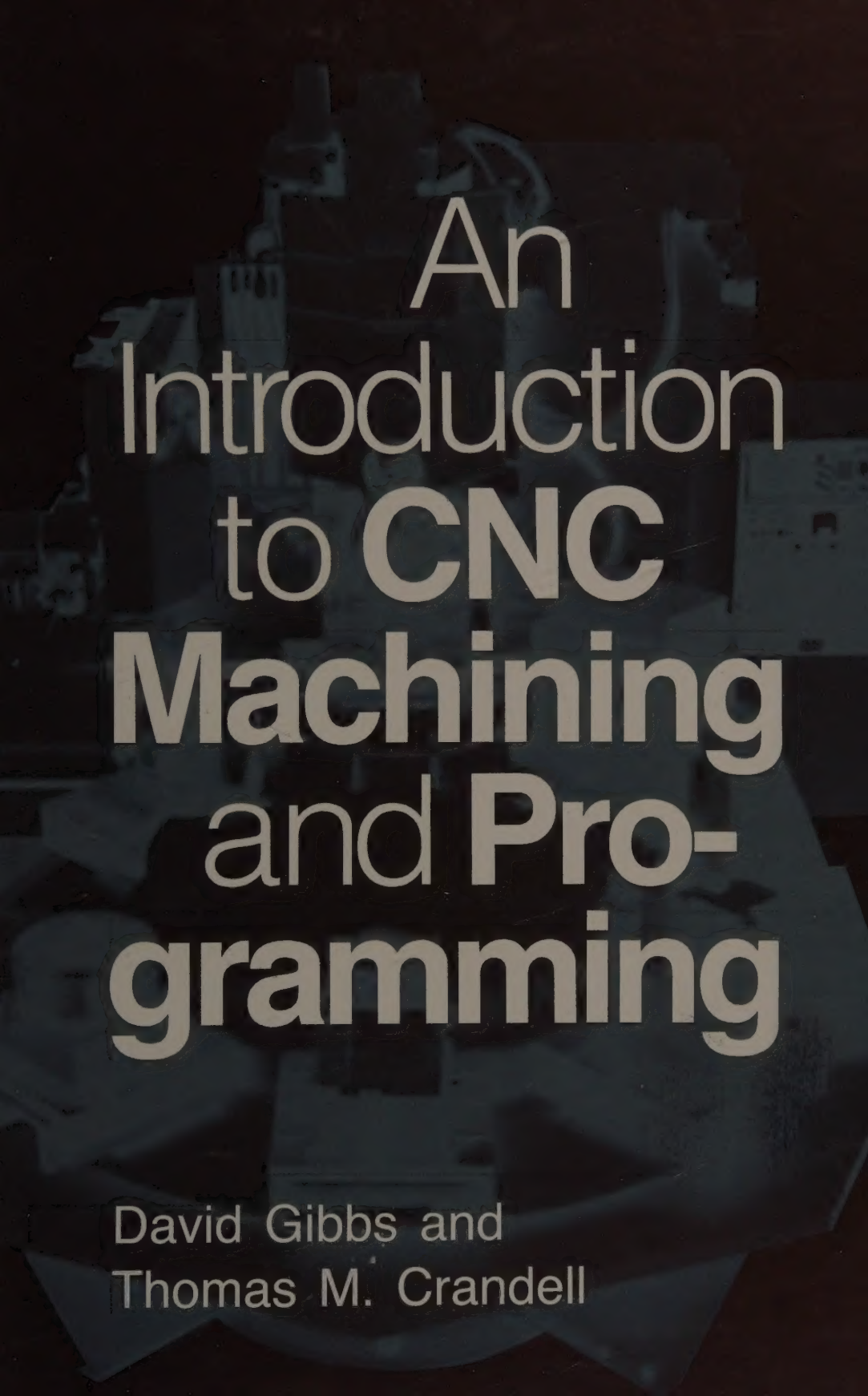




An Introduction to **CNC** **Machining** and **Pro-** **gramming**

David Gibbs and
Thomas M. Crandell

An Introduction To
**CNC Machining
and
Programming**

David Gibbs

I. Eng., MIED

Senior Lecturer in the Department of Technology
Reading College of Technology
England

Thomas M. Crandell

Computer Integrated Manufacturing Coordinator
Associate Professor
Manufacturing Engineering Technologies Department
Ferris State University

Industrial Press Inc.

DEDICATION

I would like to dedicate my work on this textbook in loving memory of my grandfather, Edgar L. Crandell. I also dedicate my work to my parents Gale and Beverly Crandell. It was these three individuals who taught me to work hard to complete a task and to do it to the best of my ability. I thank them for their time and patience during my upbringing.

My thanks goes to the following: My family—Linda, Chad, and Todd—for time spent away from them; Ferris-State University for equipment support; and Ferris Faculty and Staff that provided assistance.

Thomas M. Crandell

Portions of this text were originally published in Great Britain by Cassell Publishers Limited as *An Introduction to CNC Machining, 2nd edition*, © 1987 by David Gibbs and *CNC Part Programming*, © 1987 by David Gibbs.

Library of Congress Cataloging-in-Publication Data

Gibbs, David.

An introduction to CNC machining and programming/David Gibbs, Thomas M. Crandell.
552 p. 15.6 × 23.5 cm. Includes index. ISBN 0-8311-3009-1

1. Machine-tools—Numerical control—Programming. I. Crandell, Thomas M. II. Title.

TJ1189.G53 1991

621.9'023—dc20

90-23499
CIP

INDUSTRIAL PRESS INC.

200 Madison Avenue
New York, New York 10016-4018

Copyright © 1991 by Industrial Press Inc., New York, New York. Printed in the United States of America. All rights reserved. This book, or parts thereof, may not be reproduced, stored in a retrieval system, or transmitted in any form without the permission of the publishers.

CONTENTS

Preface

1

An Introduction to the Concept of Computer Numerical Control 1

2

Machine Design 16

3

Tooling for Computer Numerically Controlled Machining 41

4

Work Holding and Loading for Computer Numerically Controlled Machining 78

5

Data Preparation and Input to Machine Control Units 95

6

Terms and Definitions Associated with Part Programming and Machine Control 116

7

Speeds and Feeds for Numerically Controlled Machining 152

8

Part Programming for Computer Numerically Controlled Machining 162

9

Part Programming Calculations 309

10

Computer-Aided Part Programming 334

11

Advanced Techniques 436

Appendix A

EIA Specifications 460

Appendix B

Cutting Speed and Feed Information, Carbide Grades,
and Power Requirements Formulas 461

Appendix C

Programming Exercises 493

Appendix D

GDT Symbols 522

Appendix E

Formulas 525

Index 531

PREFACE

An Introduction to CNC Machining and Programming is intended to support the essentially practical activity of preparing and proving computer numerical control (CNC) part programs for turning, milling, and drilling. It will be of value to students in a wide range of courses dealing with CNC programming and calculations of all forms, tooling for CNC, and fixturing for CNC whether in a major or related course in a college, university, or industrial organization.

The preparation and proving of CNC part programs requires access to machinery and computer installations in order to obtain the necessary practical experience. Using such equipment, and understanding particular programming languages and techniques, requires instruction, examples, and exercises from a competent instructor. Students undertaking a course of study devoted to part programming will therefore find it necessary to attend an adequately resourced college or training center. The student must also have a good understanding of basic machining techniques, and should ideally have previous experience in turning, milling, and drilling operations. In preparing this text, these fundamental requirements have been borne in mind.

CNC part programming is an absorbing and time-consuming activity—it is one of the few areas of study where students complain that time has passed too quickly! Thus a primary objective of this book is to ensure that limited course time can be used to the best advantage by providing the opportunity to devote as much time as possible to preparing programs and using the associated equipment. Accordingly, an attempt has been made to include sufficient information to provide the student with much of the theoretical knowledge needed to support the more practical elements of study, thereby reducing the time spent on formal lectures and unnecessary note taking. The text also provides the student with the opportunity to study specific aspects of interest or needs.

This text is essentially practical in nature and is intended to provide adequate material for course work. It contains a series of assignments that provide the student with a practical understanding of CNC tooling, processing, and programming by various means. Throughout the book there are numerous fully detailed drawings of components in inch and metric units that, while primarily included to complement the text, may also be used as programming exercises in the early stages of a course. An additional series of projects, of varying degrees of complexity and intended for later use, should satisfy most levels of ability.

It is the author's experience that many mature people returning to college

PREFACE

for retraining, also many younger students, are hampered in their programming work by never being taught how to apply their calculation skills in algebra, geometry, and trigonometry. It is generally outside the scope of a course of study devoted to part programming to spend much time rectifying this state of affairs, and yet it cannot be ignored. To assist both instructors and students there is a chapter devoted entirely to the type of calculations that will be encountered when preparing part programs manually; it is hoped that the completion of this material, supported by on-the-spot tutoring by faculty, will be of value.

This text will be of on going value to students, faculty, and industrial programmers alike.

D.A.W. Gibbs
Workingham

Thomas M. Crandell
Ferris State University

1

AN INTRODUCTION TO THE CONCEPT OF COMPUTER NUMERICAL CONTROL

DEFINITION OF NUMERICAL CONTROL

Numerical control (NC) is the term used to describe the control of machine movements and various other functions by instructions expressed as a series of numbers and initiated via an electronic control system.

Computerized numerical control (CNC) is the term used when the control system utilizes an internal computer. The internal computer allows for the following: storage of additional programs, program editing, running of programs from memory, machine and control diagnostics, special routines, and inch/metric-incremental/absolute switchability.

The two systems are shown diagrammatically in Figure 1.1. The control units may be free-standing or built into the main structure of the machine. The operating panel of an integrated control unit is shown in Figure 1.2.

THE APPLICATION OF COMPUTER NUMERICAL CONTROL

Computer numerical control is applied to a wide range of manufacturing processes such as metal cutting, woodworking, welding, flame cutting, sheet metal forming, sheet metal punching, water jet cutting, electrical discharge machining and laser cutting. The text that follows is restricted to its application to common machine-shop engineering processes, namely, turning, milling, and drilling, where it has been particularly successful.

THE ADVANTAGES OF COMPUTER NUMERICAL CONTROL

Computer numerical control is economical for mass, batch, and, in many cases, single-item production. Many factors contribute to this economic viability, the most important of these being as follows:

- (a) high productivity rates
- (b) uniformity of the product
- (c) reduced component rejection

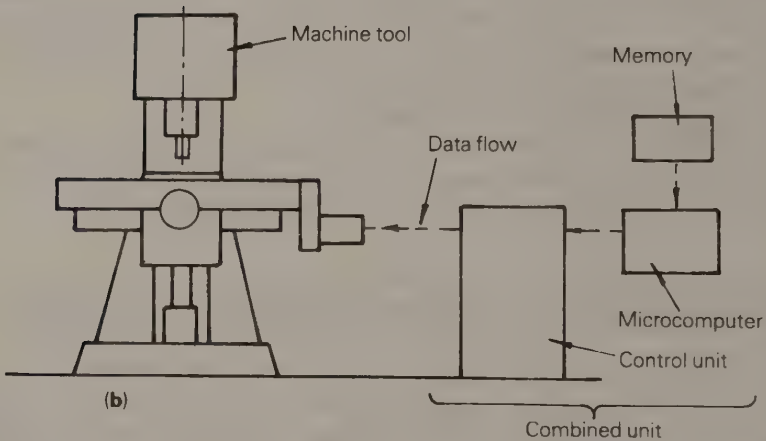
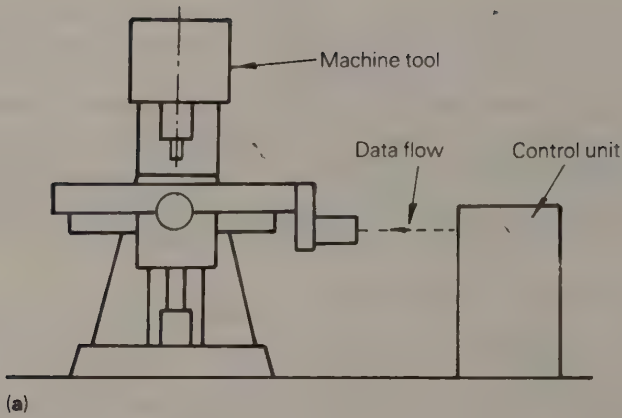


Figure 1.1 Basic control systems: (a) numerical control and (b) computerized numerical control.

- (d) reduced tooling costs
- (e) less operator involvement
- (f) complex shapes machined easily

It is also the case that fewer employees will be required as conventional machines are replaced by modern technology, but those employees that remain will of necessity be high caliber technicians with considerable knowledge of metal-cutting methods, cutting speeds and feeds, work-holding, and tool-setting techniques and who are familiar with the control systems and programming for numerical control.



Figure 1.2 *Integrated control unit.*

THE CAPABILITY OF COMPUTER NUMERICAL CONTROL

The dramatic effect computer numerical control has already had on traditional engineering production techniques is now well appreciated. Machines controlled in this way are capable of working for many hours every day virtually unsupervised. They are readily adaptable to facilitate production of a wide range

of components. Every function traditionally performed by the operator of a standard machine tool can be achieved via a computer numerical control machining program.

To appreciate just how versatile computer numerical control can be, it is only necessary to examine very briefly the human involvement in the production of a simple component such as the one shown in Figure 1.3. The hole only is to be produced by drilling on a conventional vertical milling machine. The activities of the operator in producing the component would be as follows:

1. Select a suitable cutting tool.
2. Locate the cutting tool in the machine spindle.
3. Secure the cutting tool.
4. Locate the component in the work-holding device.
5. Clamp the component.
6. Establish a datum in relation to face A.
7. Determine the amount of slide movement required.
8. Determine the direction of slide movement required.
9. Move the slide, monitoring the movement on the graduated dial allowing for leadscrew backlash, or digital readout if available.
10. Lock the slide in position.
11. Establish a second datum in relation to face B.

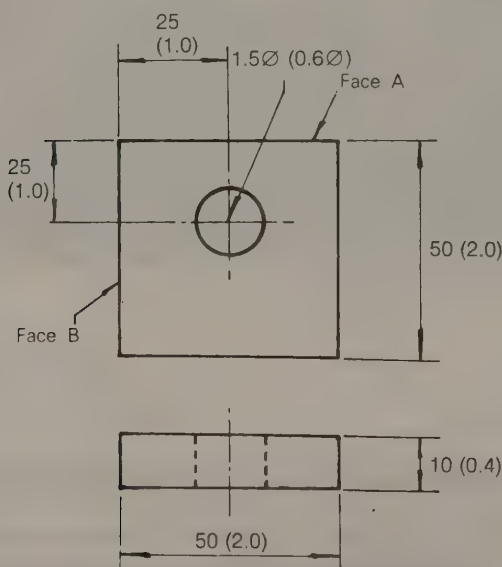


Figure 1.3 Component detail. (Inch units are given in parentheses.)

12. Determine the amount of slide movement required.
13. Determine the direction of slide movement required.
14. Move the slide, monitoring the movement on the graduated dial allowing for leadscrew backlash, or digital readout if available.
15. Lock the slide in position.
16. Select a suitable spindle speed.
17. Determine the direction of spindle rotation.
18. Select a suitable feed rate.
19. Switch on the spindle motor.
20. Switch on the coolant supply motor.
21. Engage the feed and machine the hole.
22. Disengage feed and withdraw tool.
23. Switch off the coolant supply motor.
24. Switch off the spindle motor.
25. Remove the component.
26. Verify the accuracy of the machine movement by measuring the component.

From this list it can be seen that even the simplest of machining operations involves making a considerable number of decisions that influence the resulting physical activity. A skilled machinist operating a conventional machine makes such decisions and takes the necessary action almost without thinking. Nevertheless, the decisions *are* made and the action *is* taken.

It is not possible to remove the human involvement totally from a machining process. No automatic control system is yet capable of making a decision in the true sense of the word. Its capability is restricted to responding to a manually or computer-prepared program, and it is during the preparation of the program that the decisions are made. Via that program the machine controller is fed with instructions that give effect to the decisions. In this way all the functions listed above, and many others not required in such a simple example of machining, may be automatically and repeatedly controlled. Figure 1.4 lists the elements of total machine control.

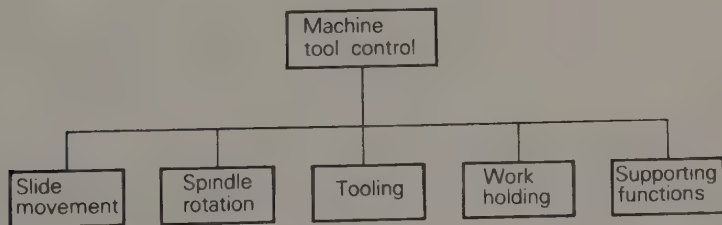


Figure 1.4 Elements of machine control.

SLIDE MOVEMENT

The success of any manual machining exercise is dependent on many factors, not least of which is the experienced worker's practical skills. These skills are most in evidence when they affect the accuracy of the finished product, such as when they are involved in positioning, via the machine slides, the cutting tool and workpiece in the correct relationship to each other. This aspect of machining skill is also the crucial factor when the machine is electronically controlled.

Slide movement on computer numerically controlled machines is achieved by:

- (a) hydraulically operated pistons
- (b) electric servo motors.

The use of electric motors is by far the most common technique. The motor is either directly coupled, or connected via a toothed belt drive, to the slide leadscrew. The servo motor, in effect, replaces the conventional handwheel and this is illustrated in Figure 1.5, which shows conventional machines, a center lathe and a vertical milling machine, fitted with servo motors. A few

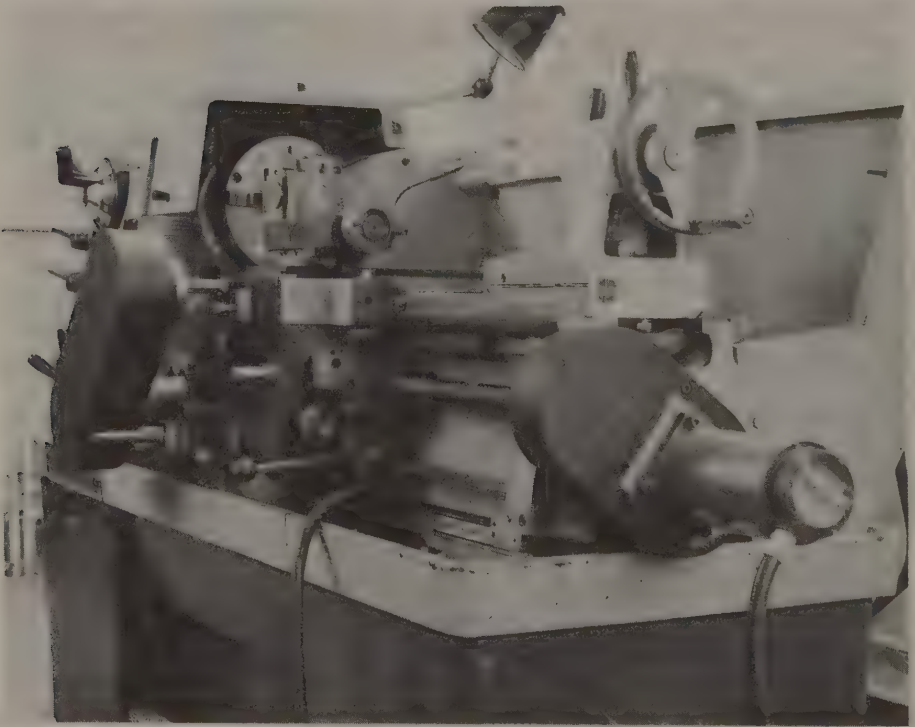


Figure 1.5 (a) *Conventional center lathe fitted with servo motors.*

machine designs have retained handwheels as an aid to setup or to provide for both numerical and manual control.

Machine tools have more than one slide and so the slide required to move will have to be identified. The plane in which movement can take place may be longitudinal, transverse, or vertical. These planes are referred to as axes and are designated by the letters *X*, *Y*, *Z*, and sometimes *U*, *V*, *W*. Rotary axes *A*, *B*, and *C* can also be applied to a machine around a center axis mentioned previously. A rotary axis has as its centerline one of the three standard axes (*X* to *A*, *Y* to *B*, and *Z* to *C*). Their location on common machine tools is shown in Figure 1.6. Note that the *Z* axis always relates to a sliding motion parallel to the spindle axis.

The direction in which a slide moves is achieved by the direction of rotation of the motor, either clockwise or counterclockwise, and the movement would be designated as plus or minus in relation to a given datum. Figure 1.6 also shows how the direction of travel is designated on common machine tools. Slide movement and relative tool and work movement are discussed in more detail in Chapter 6.

The rate or speed at which slide movement takes place, expressed in feet/meters per minute or inches/millimeters per revolution of the machine spindle,

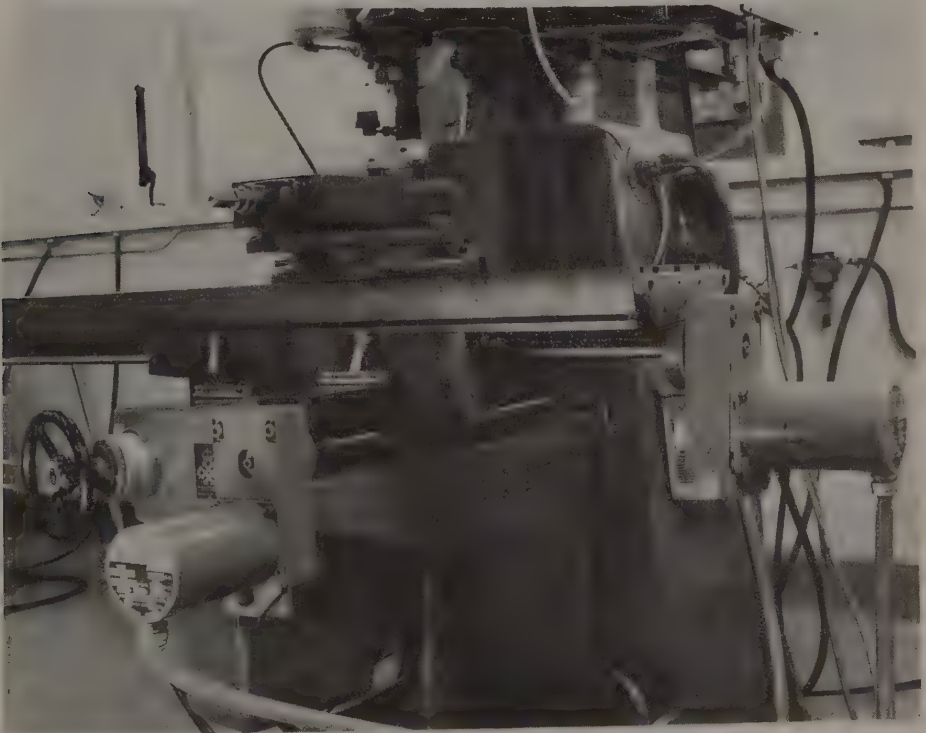


Figure 1.5 (b) *Conventional milling machine fitted with servo motors.*

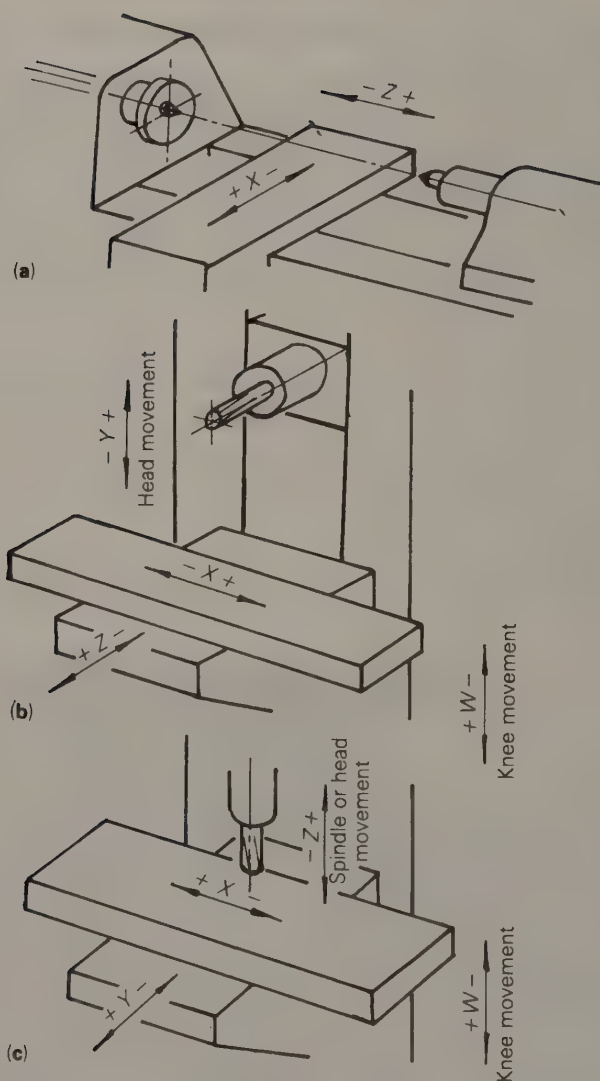


Figure 1.6 Identification of slides and direction of the slide movement on common machine tools: (a) center lathe (turning center); (b) horizontal milling machine (horizontal machining center); (c) vertical milling machine (vertical machining center).

will be proportional to the revolutions per minute of the servo motor; the higher the revolutions per minute, the faster the rate of slide travel.

The length of slide movement is controlled by either the number of revolutions or the number of part revolutions the motor is permitted to make, one complete revolution being equal to the lead of the leadscrew, in the same way as one turn of a handwheel is equal to the lead of a leadscrew. In some cases



Figure 1.7 Cogged belt drive from servo motor to leadscrew.

there may be reduction pulleys or gears between the motor and the leadscrew, as shown in Figure 1.7, in which case the linear movement obtained in relation to the motor revolutions would be proportionally reduced. The length of travel made, or required to be made, by a slide is referred to as a coordinate dimension.

Since the slide movement is caused by the servo motor, control of that motor will in turn control the slide movement. The motor is controlled electronically via the machine control unit. All the relevant information, that is the axis, direction, feed rate, and length of movement, has to be supplied to the control unit in an acceptable numerical form. The input of information to the machine controller is achieved in a variety of ways: perforated tape, magnetic tape, via a computer link, computer disk, and manually. Data input is covered in more detail in Chapter 5.

Complex Slide Movement

So far, consideration has been given to simple linear movement involving one slide. There are, however, many instances when two or more slides have to be moving at the same time. It is possible to produce a 45° angle as shown in

Figure 1.8 by synchronizing the slide movements in two axes, but to produce the 30° angle in Figure 1.9 would require a different rate of movement in each axis, and this may be outside the scope of a simple NC system unless it is capable of accurately responding to two precalculated feed rates.

Similarly, the curve shown in Figure 1.10 would present problems, since ideally its production would require constantly changing feed rates in two axes. The curve could be designated by a series of coordinate dimensions as shown

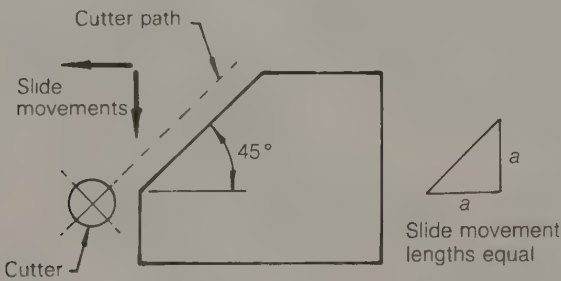


Figure 1.8 Effect of equal rates of slide movement.

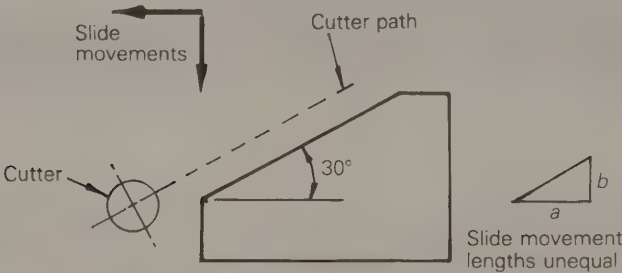


Figure 1.9 Effect of unequal rates of slide movement.

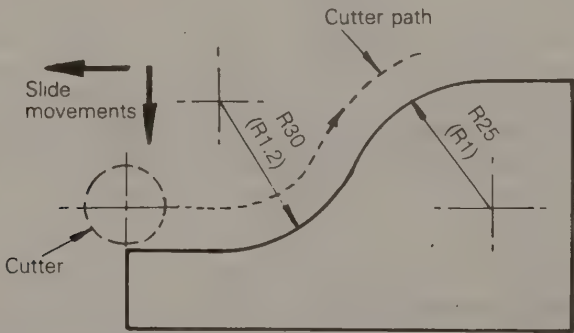


Figure 1.10 Profile requiring constantly changing rates of slide movement. (Inch units are given in parentheses.)

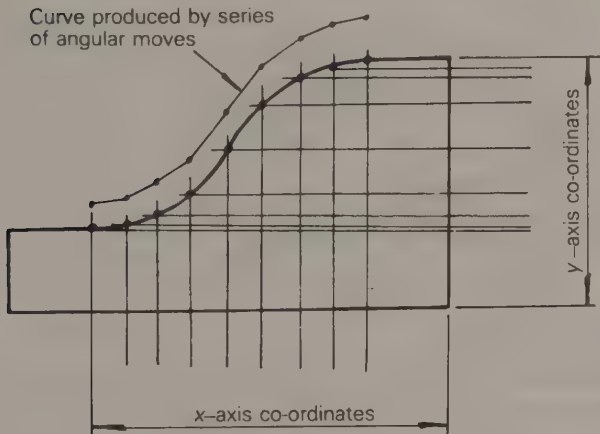


Figure 1.11 Designation of a curved profile by a series of coordinate dimensions.

in Figure 1.11, and, providing the machine were capable of responding to the minute variations in size, a satisfactory result would be obtained, but the calculations necessary to approach the task in this way would be considerable. Complex slide movements such as those required to produce the curve can readily be achieved by the inclusion in the system of a computer capable of making the necessary calculations from the minimum of input data. Of course, the calculation of slide movements to produce complex profiles is not the only function of a computer. The other facilities it provides, in particular its ability to store data that can be used as and when required, will be considered later.

Verification of Slide Movement

An important function of the skilled worker operating a conventional machine is to monitor the slide movement and verify its accuracy by measuring the component. A similar facility is desirable on computer numerically controlled machines.

Control systems without a facility to verify slide movements are referred to as "open-loop" systems, while those with this facility are called "closed-loop" systems. A closed-loop system is shown diagrammatically in Figures 1.12 and 1.13.

The exact position of the slide is monitored by a transducer and the information is fed back to the control unit, which in turn will, via the feed motor, make any necessary corrections.

In addition to positional feedback some machines are equipped with "in-process measurement." This consists of probes that touch the machined surface and respond to any unacceptable size variation. The data thus gathered are fed back to the control system and corrections to the slide movement are made automatically.

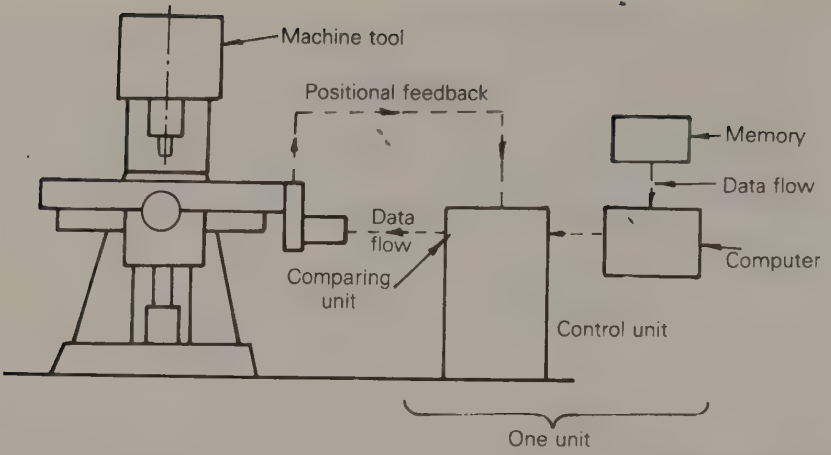


Figure 1.12 Closed-loop control system.

(Courtesy of AIMTECH.)

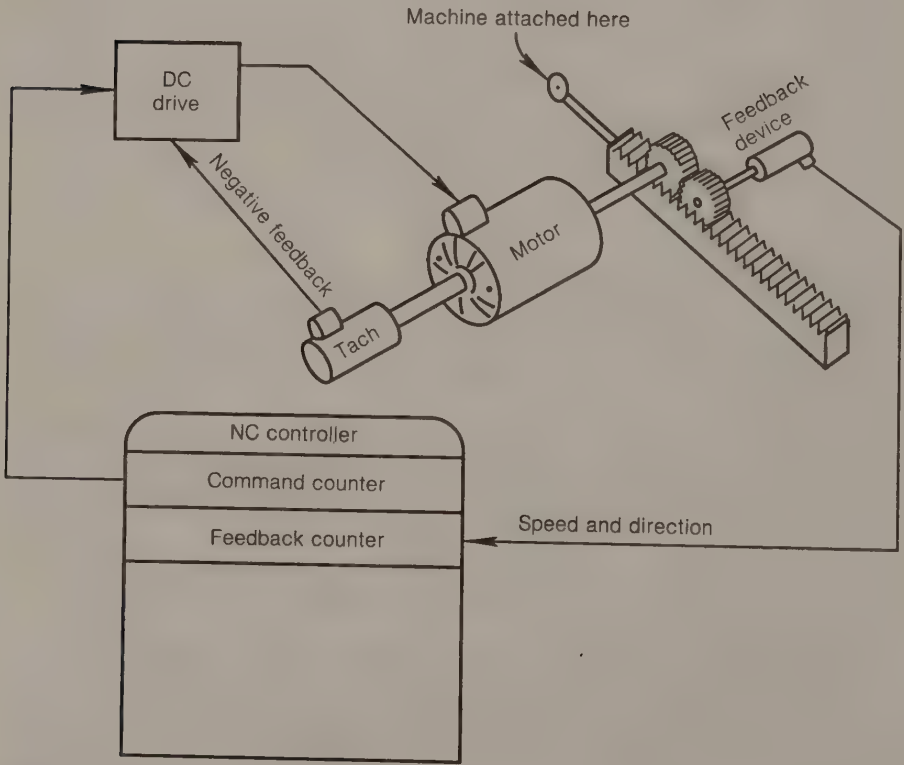


Figure 1.13 Basic NC hardware concept.

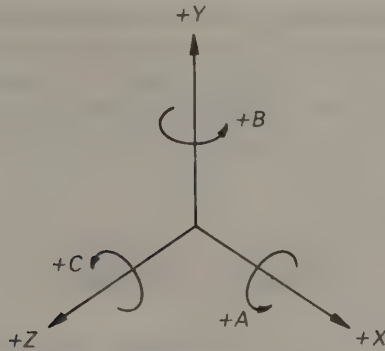


Figure 1.14 Identification of rotary movements.

ROTARY MOVEMENTS

Sometimes the production of a component requires rotary movement in addition to the linear movement of the machine-tool slides. This movement is provided by ancillary equipment such as rotary tables and indexers. These movements are controllable via the machining program. They are identified by the letters A, B, and C as indicated in Figure 1.14.

CONTROL OF MACHINE SPINDLES

Machine spindles are driven directly or indirectly by electric motors, and a few by hydraulic drive. The degree of automatic control over this motion usually includes stopping and starting, and the direction and speed of rotation. Some very early systems, and perhaps a few inexpensive modern systems, do not include control of the spindle motions at all, switching on and off and gear selection being a totally manual operation. On the other hand, on some very modern control systems the torque or horsepower necessary to carry out the machining operation can be monitored and compared with a predetermined value included in the machining program; when necessary, the spindle speed will be varied automatically to provide optimum cutting conditions. (See "Adaptive Control," Chapter 9.)

The speed of the spindle is often infinitely variable, and may automatically change as cutting is taking place to maintain a programmed surface speed. Thus, when facing the end of a bar on a lathe as the tool nears the work center, the spindle speed will increase. In this way material removal is achieved at the fastest possible rate with due regard to tool life and the surface finish required.

The direction of spindle rotation required can be determined as follows:

1. Clockwise (CW). When the spindle rotates a right-handed screw would

advance into the workpiece, or if the machine operator looked through the tool toward the workpiece, he would see it moving clockwise.

2. Counterclockwise (CCW). When the spindle rotates a right-handed screw would retract from the workpiece, or if the machine operator looked through the tool toward the workpiece, he would see it moving counterclockwise.

CONTROL OF TOOLING

Computer numerically controlled machines may incorporate in their design turrets or magazines that hold a number of cutting tools. The machine controller can be programmed to cause indexing of the turret or magazine to present a new cutting tool to the work or to facilitate tool removal and replacement where automatic tool-changing devices are involved.

Simpler machines rely on manual intervention to effect tool changes. In these cases the control unit is programmed to stop the automatic sequence at the appropriate time and the operator will make the change. There is sometimes a connection between the control unit and the tool-storage rack and the correct tool to be used is indicated by an illuminated lamp.

Tooling is dealt with in more detail in Chapter 3.

CONTROL OF WORK HOLDING

Work holding is another aspect of computer numerically controlled machining that can include manual intervention or be totally automatic. The work-holding devices themselves can be fairly conventional: vices, chucks, collets, and fixtures are all used. The computer numerical control can extend to loading the workpiece by the use of robots and securely clamping it by activating hydraulic or pneumatic clamping systems.

Again, as with tool changing, on simpler machines, a programmed break in a machining cycle can facilitate manual intervention as and when required.

Work holding is dealt with in detail in Chapter 4.

SUPPORTING FUNCTIONS

The various supplementary functions a skilled worker would perform during a manually controlled machining operation are, of course, vital to the success of the operation. For example, it may be necessary to clamp a slide, apply coolant, clear away swarf before locating a component, monitor the condition of tooling, and so on. Slide clamping is usually hydraulic, and hydraulic pressure provided by an electrically driven pump with the fluid flow controlled by so-

lenoid valves has long been a feature of machine tool design. With the new technology the control of the electrical elements of such a system is included in the machining program. Similarly, it is a simple matter to control the on-off switching of a coolant pump and the opening or closing of an air valve to supply a blast of cleaning air. Tool monitoring, however, is more complex and is the subject of much research and innovation ranging from monitoring the loads exerted on spindle motors to recording variations in the sound the cutting tool makes. Some of these more advanced features of computer numerical control are discussed further in Chapter 9.

QUESTIONS

- 1 Explain with the aid of a simple block diagram the difference between an NC and a CNC machining system.
- 2 State two advantages of CNC over NC control systems.
- 3 The common axes of slide movement are X, Y, and Z. What is significant about the Z axis?
- 4 How are rotary movements about a given axis identified and when are they likely to be used?
- 5 What data are required to initiate a controlled slide movement?
- 6 On a vertical machining center the downward movement of the spindle is designated as a Z minus. From a safety aspect this is significant. Why is this so?
- 7 How is an angular tool path achieved?
- 8 With the aid of simple block diagrams to show data flow, explain the difference between an open-loop and a closed-loop control system.
- 9 How would a manual tool change be accommodated in a machine program?
- 10 Explain what is meant by "constant cutting speed" and how this is achieved on CNC machines.

2

MACHINE DESIGN

REPEATABILITY

The quality of conventional machine tools varies considerably. They are built to a price to meet a wide-ranging market. Generally speaking, the more expensive the machine is, the higher the quality of work that can be expected to be produced on it. However, an expensive conventional machine does not guarantee high-quality work. The key to success lies in the skills of the operator. The cheapest of machines is capable of producing very accurate work in the hands of the right person.

Skilled workers get to know their machines and make allowances for their failings. During the production of a component a skilled worker can, for example, compensate for leadscrew backlash, slide friction, lack of power, and so on. He or she can vary spindle speeds, feed rates, and tooling arrangements. The approach to a final cut can be gradual until it is correct and before a final commitment is made.

With a computer numerically controlled machine tool responding to a pre-determined program, the capacity for readily varying the conditions when machining is under way is limited, and to make changes is inconvenient. As far as possible conditions have to be correctly determined at the time the program is produced and the machine is set up.

The slide movements are of prime importance. The movement must be precise, and this precision must continue throughout a machining program, which may involve thousands of components. The ability of the machine to produce continually accurate slide movement is called repeatability.

A precise definition of repeatability is as follows: the maximum difference that can occur between the shortest and longest positions achieved in a number of attempted moves to any programmed target position.

Repeatability is expressed as the mean of a number of attempted moves. A typical figure for repeatability would be ± 0.0003 in. or ± 0.008 mm. It follows that some moves must be well within those figures.

Repeatability is dependent on the following features being incorporated in the design of the machine:

- (a) adequate strength
- (b) rigidity
- (c) minimum of vibration

- (d) dimensional stability
- (e) accurate control of the slide movements

Although many conventional machines have been, and continue to be, converted to computer numerical control, such conversions being referred to as “retrofits,” their design in general does not meet the exacting requirements necessary to achieve a high standard of repeatability, while at the same time catering to the needs for high rates of metal removal that modern tooling and electronic control have made possible. Radical changes in design were inevitable and have resulted in the machines now generally known as vertical machining centers, horizontal machining centers, and turning centers. These are shown in Figure 2.1.

STATIC AND DYNAMIC LOADING

A simple analysis of the function of a machine tool reveals that it is subjected to certain loading which may be described as:

- (a) static
- (b) dynamic

Static loading is the term used to describe a situation where forces are acting on a structure when the machine, or that part of the machine, is not in motion.

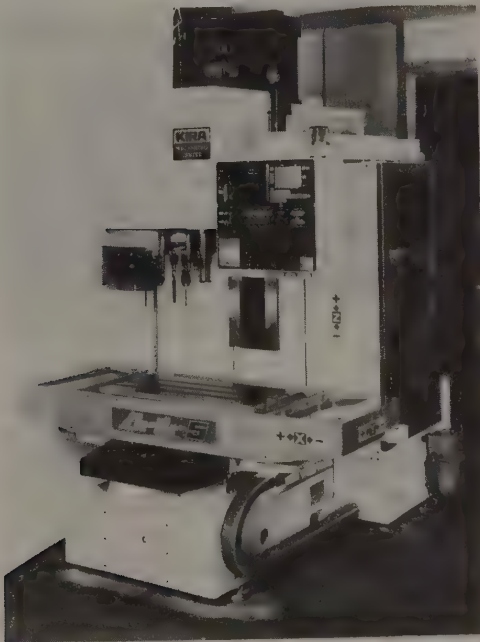


Figure 2.1 (a) *Vertical machining center.*

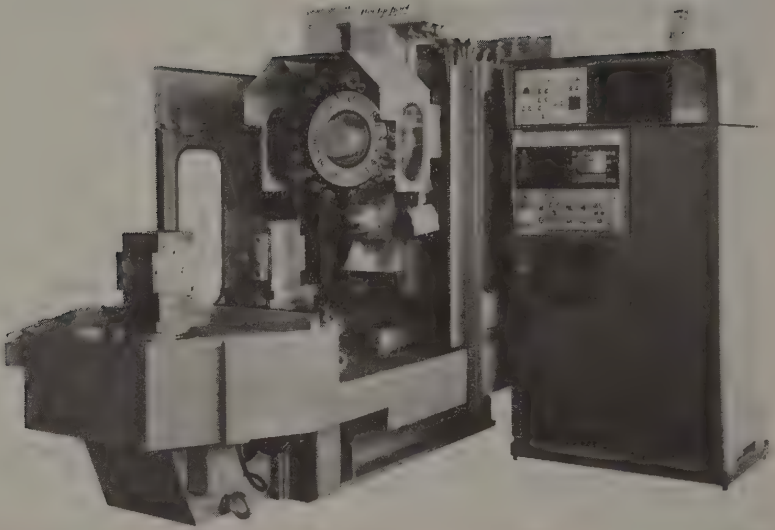


Figure 2.1 (b) *Horizontal machining center.*

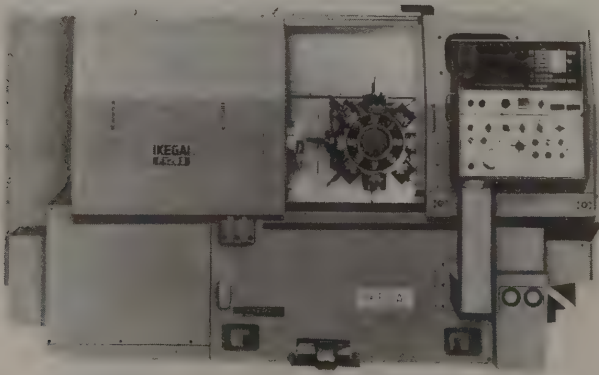


Figure 2.1 (c) *Turning center.*

For example, due to its mass, a milling machine table exerts a static load on the knee. If the table is offset on the knee, that static load could cause the table to drop slightly at the unsupported end. A heavy workload would exacerbate the problem, which is illustrated in Figure 2.2.

Dynamic loading is the term used to describe a situation where forces are acting on a structure when movement is taking place. An example of this, shown in Figure 2.3, is the radial force exerted on a milling machine spindle as the cutter is fed into the work. The spindle could deflect.

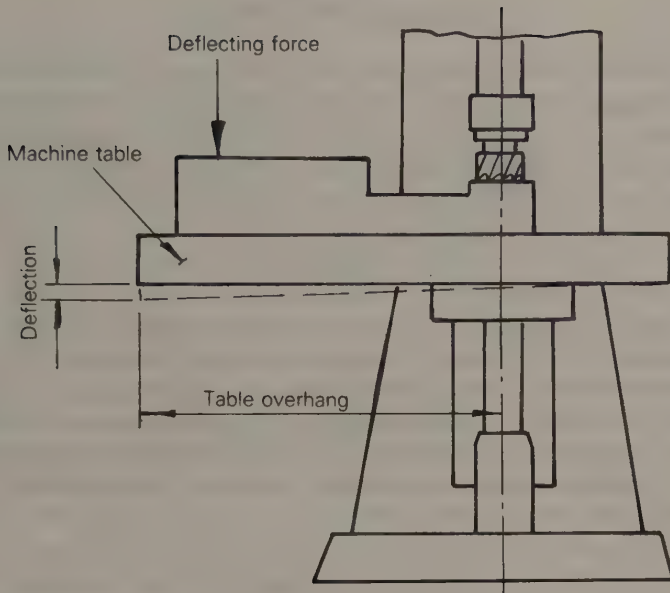


Figure 2.2 Example of the possible effect of static loading.

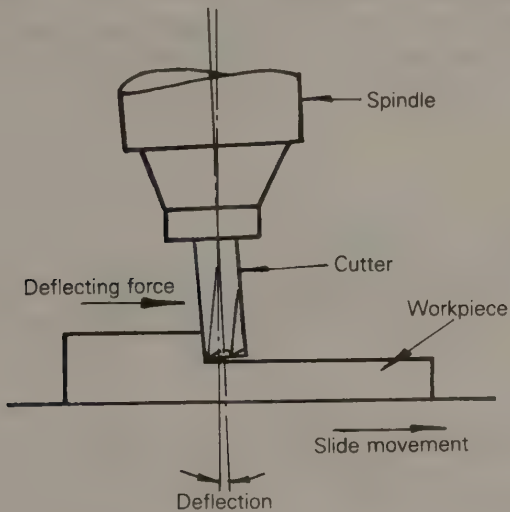


Figure 2.3 Example of the possible effect of dynamic loading.

Deflections such as those illustrated need only be quite small to affect the dimensional accuracy of the workpiece, so the machine structure and its sub-assemblies must be so designed to ensure that movement of this nature cannot occur.

BASIC STRUCTURE

For many years cast iron was considered to be the only material suitable for the basic structure of a machine tool. It possessed adequate strength and rigidity and tended to absorb vibration. In addition, the complex shapes required were easier to produce by casting than by any other method. Cast iron is still extensively used, but its traditional position as the most suitable material is now challenged by steel and, more recently, by concrete.

When castings are used, they are generally of one-piece box construction, heavily ribbed and stabilized by heat treatment.

Fabricated steel structures are increasingly being favored for very large machines. Steel plates of the same thickness as a cast iron structure have approximately twice the strength. By reducing the plate thickness, the weight of the structure can be considerably reduced, yet still provide the necessary strength. In use, the rigidity of such structures has proved to be more than adequate. However, the general use of steel is limited by the problems of making complex shapes and by its resonant quality, which is not conducive to effective damping of vibration.

The use of concrete or ceramics as a machine base is a comparatively new development. The advantages of concrete are its low cost and good damping characteristics. Very large structures can be cast on site, thus reducing the overall cost even further, since no transport is involved. Smaller structures can be provided with steel tubing cast into the concrete to permit easier handling. The cast iron bed of the machine is set on a cushion of air-setting resin and attached to the concrete by steel studs. The diagram in Figure 2.4 illustrates the concept.

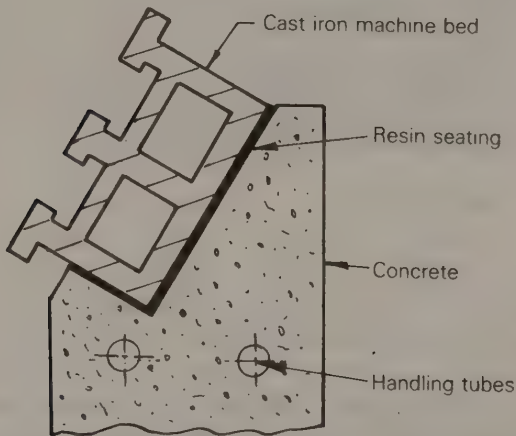


Figure 2.4 Concrete base for a machine tool.

MACHINE SPINDLES

The machine spindle is a very important design feature. The possibility of deflection has already been noted. In addition to the radial loads that cause deflection, a spindle assembly is also subjected to a thrust load acting along its axis. The design of the spindle assembly must be such that these loads are adequately contained. Inadequate support results not only in dimensional inaccuracies but also in poor surface finish and chatter. A well-supported spindle assembly is shown in Figure 2.5. Note that the spindle overhang has been kept to a minimum, a common feature of turning and horizontal machines.

The spindle of vertical machining centers presents additional problems, since it is a traditional feature of this type of machine for the spindle to move up and down. Obviously, the more the spindle is extended, the greater the risk of deflection. Some manufacturers have now moved away from the moving-spindle concept and instead the whole head assembly moves up and down.

Another design feature problem of vertical machining centers is that in order to provide an adequate work area, the spindle head must overhang. The length of overhang must be kept to a minimum, and Figure 2.6 shows how one manufacturer has improved on the traditional design without reducing the work area.

The forces that cause deflection of the spindle also result in a tendency for the complete spindle-housing assembly to twist. This has resulted in an increased use of bifurcated or two-pillar structures where the spindle housing is

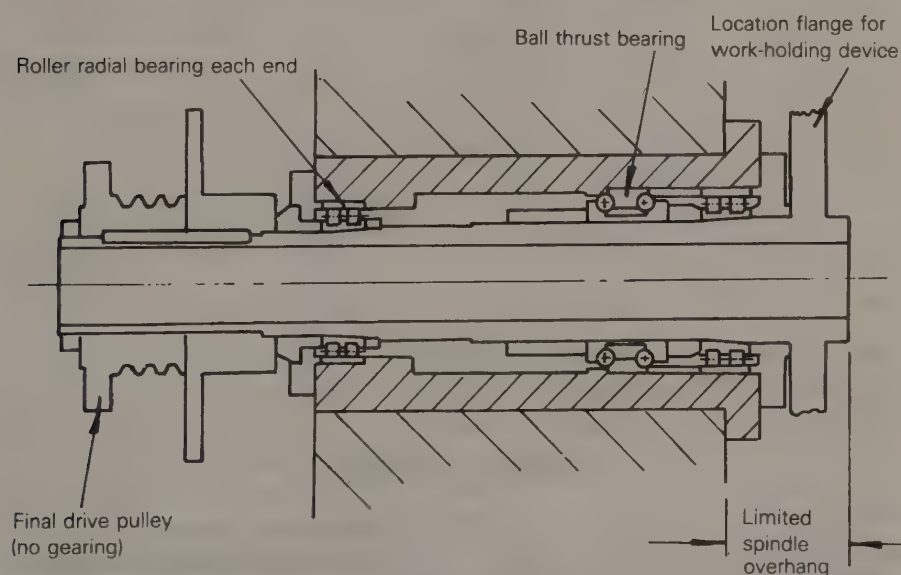


Figure 2.5 Spindle assembly for turning center.

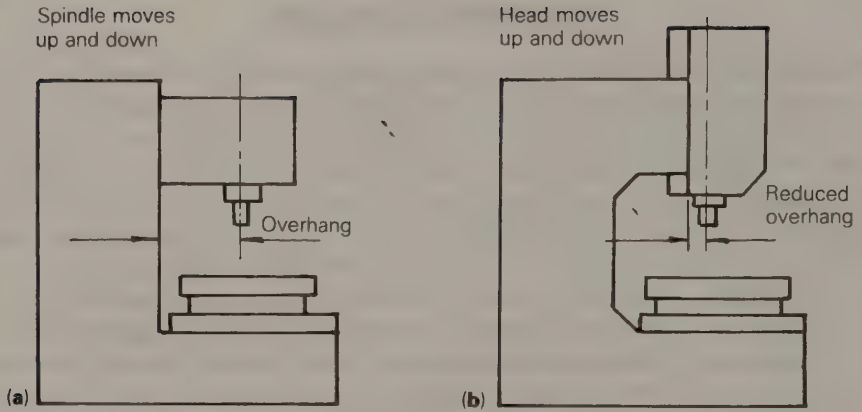


Figure 2.6 Variations in the design of vertical machining centers: (a) conventional design; (b) improved design.

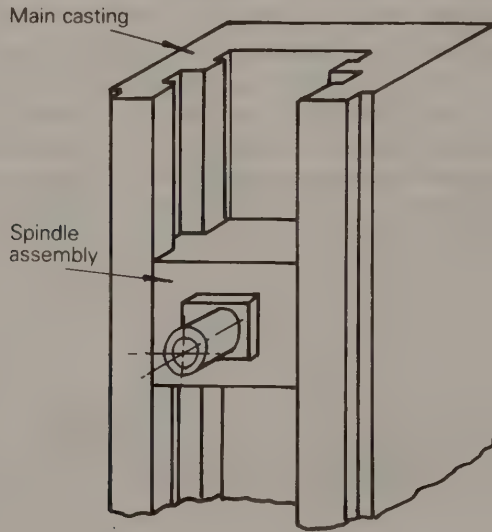


Figure 2.7 Bifurcated structure.

located between two substantial slideways that reduce the tendency to twist. A bifurcated structure is shown in Figure 2.7.

SPINDLE DRIVES

Two types of electric motors are used for spindle drives: direct current (DC) and alternating current (AC). They may be coupled direct to the spindle or via belts and/or gears. Many machines have a final belt drive which is quieter and produces less vibration than a geared drive.

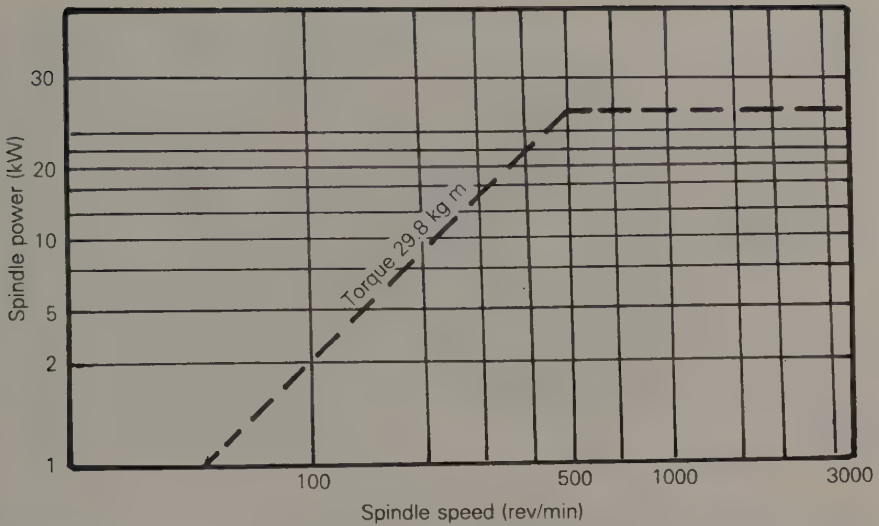


Figure 2.8 An example of torque/spindle speed relationship when driven by a DC motor: constant torque after 500 rev/min.

The majority of modern machines use DC motors. By varying the voltage input, their speeds are infinitely variable as they rotate and so a constant cutting speed can be maintained. The torque available from a DC motor is constant throughout most of the speed range, as illustrated in Figure 2.8.

There are some machines fitted with specially designed AC motors that also provide for variable spindle speeds, but the use of AC motors usually involves a stepped drive, that is, a series of spindle speeds will be available and the selection of a particular speed may involve switching from one speed range to another, high or low, for example, a feature that is common to many conventional machines. On computer numerically controlled machines the switching will be carried out as and when programmed via the control unit and may also include an automatic engagement or disengagement of an electrically operated clutch.

LEADSCREWS

The Acme form of leadscrew used on conventional machines has not proved to be satisfactory for numerically controlled machines. The movement of an Acme screw is dependent on there being clearance, i.e., backlash, between two flanks. At the same time friction between the mating flanks of the screw means that considerable resistance to motion is present. These two disadvantages are illustrated in Figure 2.9.

Computer numerically controlled machines, except perhaps for a few cheaper training machines, are fitted with recirculating ballscrews, which replace slid-

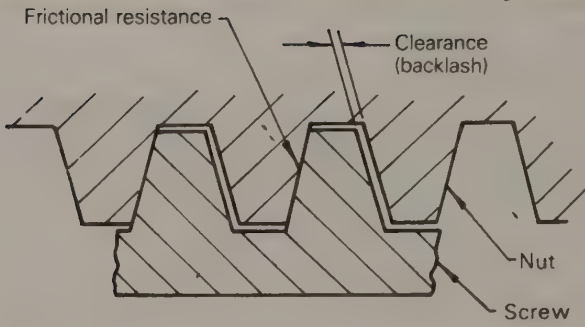


Figure 2.9 Disadvantages of conventional Acme leadscrews.

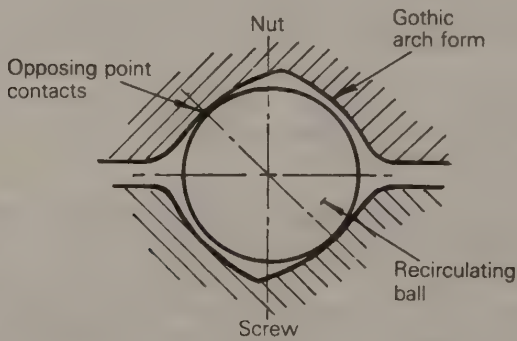


Figure 2.10 Ball screw form.

ing motion with rolling motion, resulting in reduced frictional resistance. The balls, which in effect form the nut, recirculate in and out of the thread. The thread form is referred to as a Gothic arch and is illustrated in Figure 2.10. The balls make opposing point contact which virtually eliminates backlash. Figure 2.11 shows an external ball return and Figure 2.12 an internal return. The internal ball return is more compact.

The advantages of recirculating ball screws over Acme screws are:

- (a) longer life
- (b) less wear
- (c) low frictional resistance
- (d) less drive power required due to reduced friction
- (e) higher traversing speeds can be used
- (f) no stick slip effect
- (g) more precise positioning over the total life of the machine

Leadscrews are usually of substantial diameter and centrally positioned to avoid twisting the slide and thus reducing the efficiency of the movement.

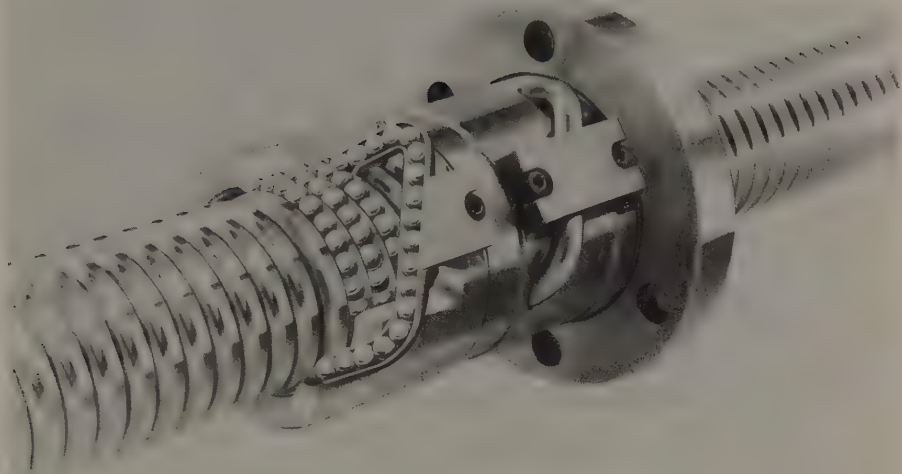


Figure 2.11 *Recirculating ball screw (external return).*

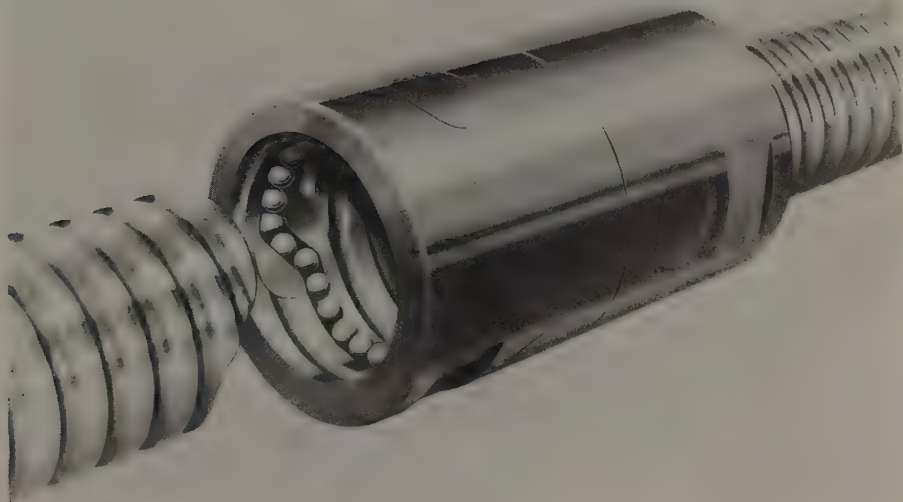


Figure 2.12 *Recirculating ball screw (internal return).*

MACHINE SLIDES

The movement of a machine slide must be smooth and responsive. There must be the minimum of frictional resistance to motion. In addition, wear, with its resulting dimensional inaccuracies, must be kept to a minimum.

The slides of a large number of computer numerically controlled machines have flat bearing surfaces. These surfaces are usually hardened and ground and coated with polytetrafluorethylene (PTFE). This surface has a low coefficient of friction, is slightly porous, and therefore is lubricant retaining. The load-bearing properties of flat surfaces are superior to those of other types of slides.

Where the machine loading permits, the sliding action of a flat bearing is sometimes replaced by a rolling action, in the form of balls or rollers, resulting in a marked reduction in the frictional resistance and requiring less power to achieve movement. Ball bushes, which may be circular or split, are illustrated in Figure 2.13. Figure 2.14 shows the practical application of split bushes. As motion takes place the balls recirculate. A similar slide arrangement involving recirculating rollers is illustrated in Figure 2.15.

Also used, but not very widely, are hydrostatic bearings where the bearing surfaces are always separated by oil or air supplied under pressure.

The automatic forced lubrication of slides is common, and protection is provided by telescopic or accordion covers.

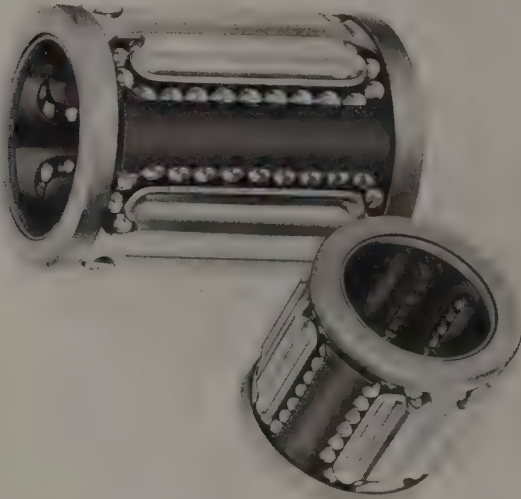


Figure 2.13 (a) *Ball bush.*



Figure 2.13 (b) *Split ball bush.*

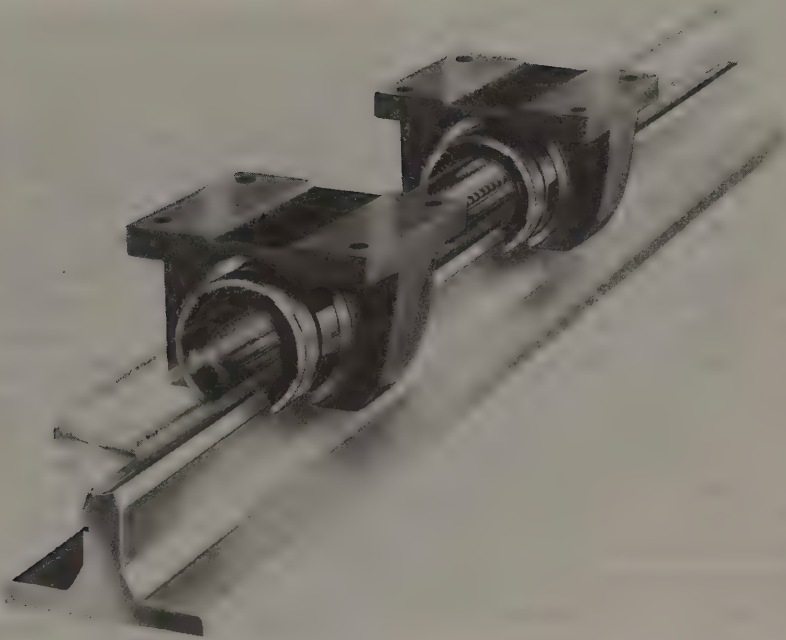


Figure 2.14 *Application of split ball bushes.*

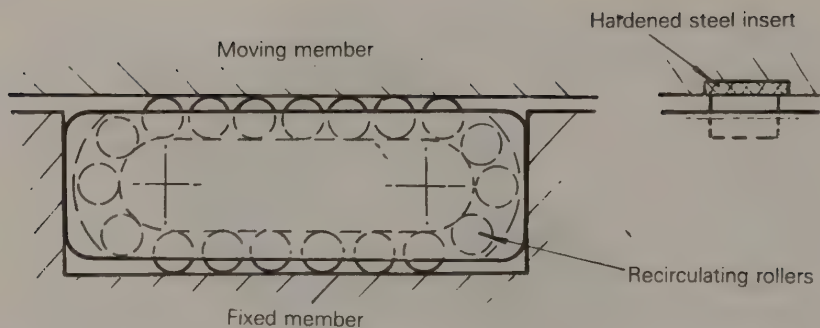


Figure 2.15 *Recirculating roller slide.*

SLIDE DRIVES

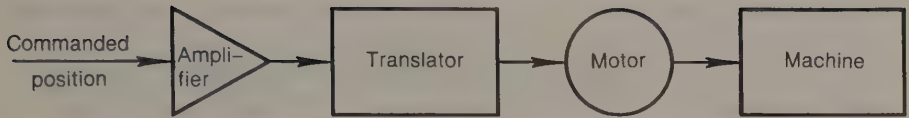
It was stated in Chapter 1 that both electric and hydraulic power are used to achieve slide motion. There are a number of very effective, responsive, and thoroughly proved hydraulic systems currently in use, but by far the most common power source is the electric motor, and so the text will be confined to dealing only with this method.

Two types of DC motor are used:

- (a) stepping motors
- (b) conventional, nonstepping motors

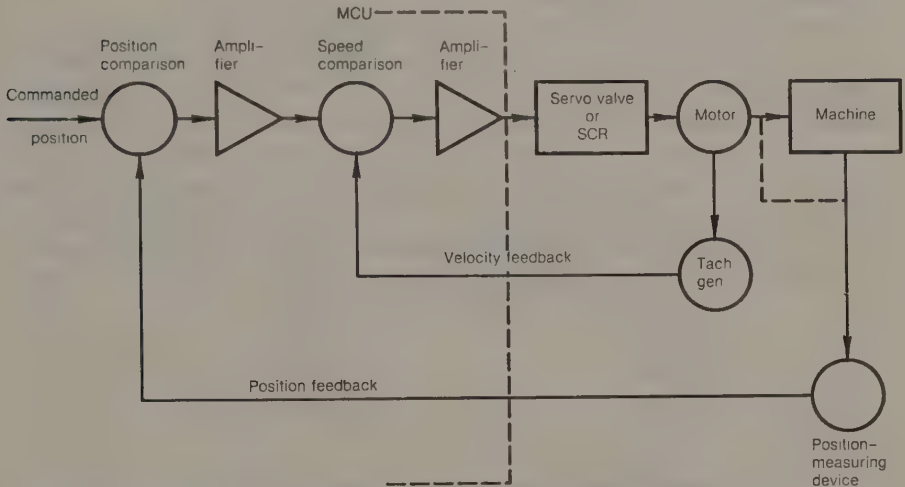
Stepping motors are a special type of motor designed so that they rotate in sequential finite steps when energized by electrical pulses. Open-loop stepping motor drive systems have two major limitations: (1) there are limited horsepower and torque ratings to meet requirements; (2) the increment size-versus-slide velocity requirements needed. An example would be an 8000 pulse per second stepper motor with a system requiring 0.0001 in. slide accuracy would have a velocity of 48 in. per minute. Therefore, the higher the machine slide accuracy required, the slower the feedrate obtainable. See open-loop servo control diagram in Figure 2.16a. This type of motor was fitted to the earlier generation of machines but has now been largely superseded by the closed-loop servo drive system, which in recent years has been the subject of much research, resulting in vastly improved designs that, together with improvements in control systems technology, make them much more responsive and easier to control than open-loop stepping motor systems.

The speeds of DC motors are infinitely variable. Constant torque is available throughout most of the speed range, which means that relatively small motors can be used, and when they are directly coupled to the machine leadscrew a torsionally stiff drive is provided. The motors provide regenerative braking, resulting in a virtually nonexistent slide overrun.



(From *Fundamentals of Numerical Control*, Publication SD-100, Allen Bradley Corp., Milwaukee, WI.)

Figure 2.16a Open-loop servo control (block diagram).



(From *Fundamentals of Numerical Control*, Publication SD-100, Allen Bradley Corp., Milwaukee, WI.)

Figure 2.16b Closed-loop servo system, servo block diagram.

The closed-loop system (Figure 2.16b) is not drive-motor-dependent using stepper motors, AC motors, DC motors, SCR drives, hydraulic motors, or hydraulic cylinders. With this type of drive system, resolutions of 50 millionths and speeds higher than 400 inches per minute are possible.

Considerable research is being carried out with AC servo motors. At present they are larger than DC motors providing equivalent power, and are also more costly. However, they need less maintenance and this is a factor very much in their favor.

POSITIONAL FEEDBACK

In Chapter 1 reference was made to the concept of open-loop and closed-loop slide positioning systems. The closed-loop positioning system is an important feature of any good computer numerically controlled machine. The concept can be summarized as instruction-movement-information-confirmation. The crucial feedback information is provided by a transducer.

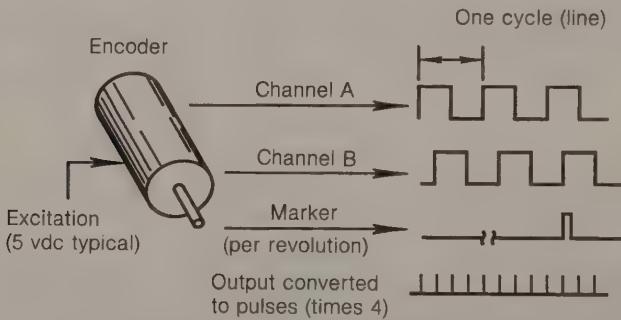
A transducer can be described as a device that receives and transmits infor-

mation. This information is received in one form, converted, and then transmitted in another form acceptable to the receiver.

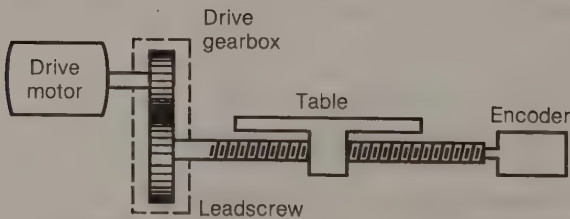
A variety of transducers have been applied, with varying success, to computer numerically controlled machines. Two of the more common types are described below.

Rotary-Type Transducers

A rotary-type transducer transmits angular displacement as a voltage. The transducer creates electrical pulses and transmits them back to the control by use of electrical windage in resolvers/synchro resolvers or photoelectric disk encoders. Physically, this transducer is attached to one end of the leadscrew either by direct means or through the use of precision gears (Figures 2.17a and 2.17b). Within this relatively small package there are a series of electrical windings. One of these windings, referred to as a rotor, rotates with the lead-screw. Around the periphery of the leadscrew are a series of interconnected windings that do not rotate and that are referred to as the stator.



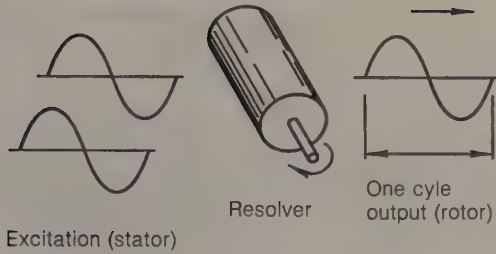
Using channels A or B or both permits encoder signals to be 1, 2, or 4 times the number of lines; marker signal occurs once per revolution and is used for reference



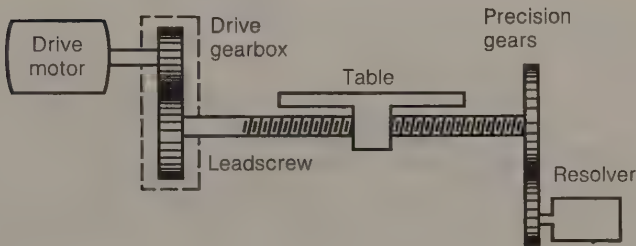
Encoder feedback coupled directly to leadscrew

(From *Fundamentals of Numerical Control*, Publication SD-100, Allen Bradley Corp., Milwaukee, WI.)

Figure 2.17a Encoder.



Output phase angle indicates shaft-angle position of the rotor



Precision gearing in the proper ratio must be used when the feedback device is not compatible with leadscrew pitch

(From *Fundamentals of Numerical Control*, Publication SD-100 Allen Bradley Corp., Milwaukee, WI.)

Figure 2.17b. Resolver.

The stator windings or photocells are fed with electrical power at a voltage rate that has been determined by the machine control unit in response to digital information relating to the required slide movements it has received via the part program. As the servo motor rotates the leadscrew, a voltage is induced in the rotor photocells, and this voltage will vary according to the angular position of the leadscrew in relation to the stator windings or photo encoder disk. Information relating to the induced voltage is fed back to the control unit, which, in effect, counts the number of complete revolutions and part revolutions the leadscrew has made, thus confirming that the movement achieved corresponds to the original instruction.

Optical Gratings

An optical grating transducer transmits linear movement as a voltage signal in the form of a series of pulses.

The principle of the optical grating can be shown in a practical way as follows. Figure 2.18 represents a pair of optical gratings, each consisting of a

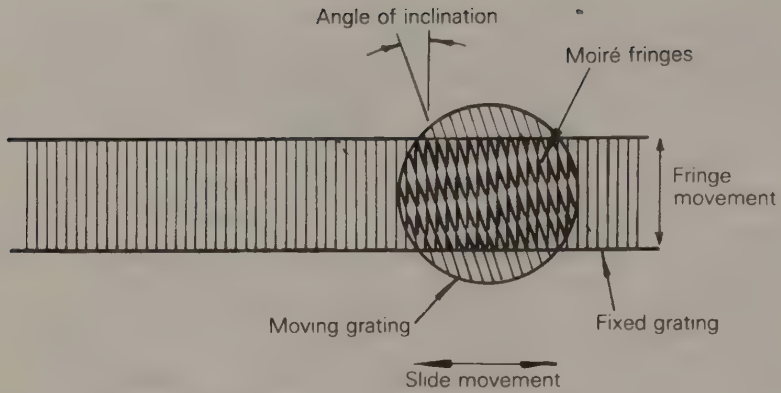


Figure 2.18 Principle of optical grating.

number of evenly spaced parallel lines. One grating is fixed and the other is caused to move along. The reader is invited to reproduce the moving grating on either a piece of tracing paper or clear plastic film. Place the second grating over the first and, with the lines inclined at a slight angle, move the second grating across the first. A fringe pattern similar to the one shown will be observed moving across the fixed grating. This pattern is referred to as Moiré fringe.

There is a mathematical connection between the spacing of the gratings, the angle of inclination and the apparent fringe movement. This principle is applied to measuring the movement of machine slides.

For practical purposes the gratings are etched either on gelatine-coated glass or on stainless steel. The fixed grating is attached to the main casting of the machine and the moving grating is positioned immediately above the fixed grating, but is attached to the moving slide. Gratings for applications of this nature have 100 to 200 spacings per inch or 25 mm.

If glass is used, a light source is directed through the grating; if stainless steel is used, the light is reflected off the surface. This light is focused onto a phototransistor, which responds according to whether the projected light is uninterrupted or interrupted, that is, a fringe is present or not present. The electrical pulses produced in this way each represent a known linear value. The number of pulses is counted and this information is fed back to the control unit as confirmation that the correct movement has been made.

Both the transducers described have a weakness. One monitors revolutions of the leadscrew, the other movements made by a slide. Neither of these factors may be a precise indication of the position of the tool in relation to the work, which would be the function of a perfect transducer. Such a transducer poses many design problems and has yet to be developed.

The locations of a linear and a rotary transducer on machine tools are illustrated in Figure 2.19.

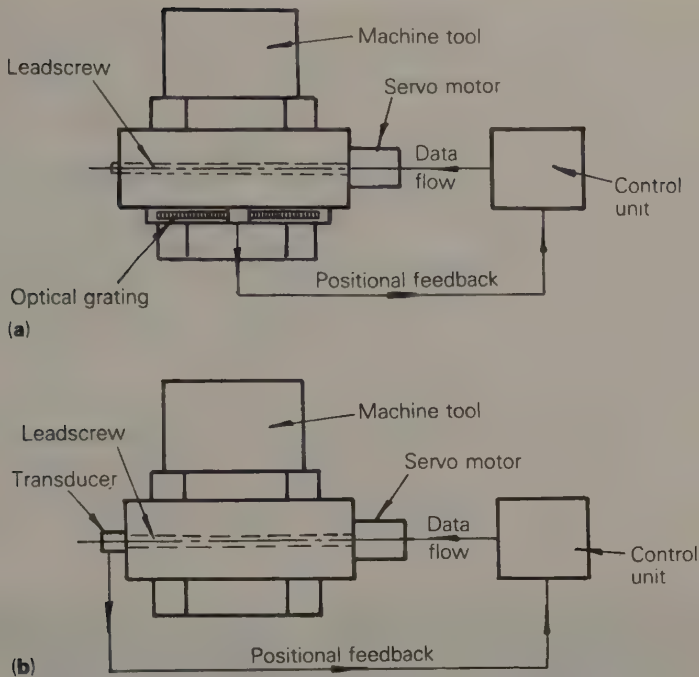


Figure 2.19 Location of positional transducers on machining centers: (a) linear transducer; (b) rotary transducer.

TEMPERATURE VARIATIONS THAT AFFECT DIMENSIONAL STABILITY

Manufacturers design their machines so that the correct alignments are achieved at a stated temperature. "Warm up" times are often quoted and sometimes warning lights are built into the control system. Deviation from the stated temperature can cause twisting or distortion of the machine castings and can have a considerable effect on the accuracy of the work produced.

Heat sources that have to be accommodated in the design of the machine, or otherwise eliminated, are as follows:

- (a) heat due to friction in motors, bearings and slides
- (b) heat due to the metal-cutting action
- (c) heat due to accumulated chips or swarf
- (d) heat in the environment

Heat due to friction is eliminated or its effect reduced in a variety of ways. For example, main drive motors are sometimes placed outside the main structure (which also helps to reduce vibration) and the final drive to the spindle is via belts. Motors that are attached to the machine body have heat radiation

facilities in the form of vanes built into their structure. Some are cooled by a ducted air flow. Spindles may be air or oil cooled; sometimes when oil is used there are cooling facilities for the recirculating oil. Heat produced by slide movement is virtually eliminated by the efforts made to produce frictionless slides, as mentioned earlier.

Heat produced by the cutting action is kept to a minimum by ensuring that the correct cutting conditions prevail, that is, by using tools with the correct geometry for the material being cut and the operation being carried out, and by ensuring that the correct cutting speed and feed rates are employed. In addition, coolant, as a flood or as spray mist, can be applied to the cutting area. Some machines provide each cutting tool with an individual coolant supply, as illustrated in Figure 2.20.

Heat due to chip accumulation can be a major problem, especially when the machines are totally enclosed for safety. The ideal situation is to have the chips falling away from the machine. Figure 2.21a illustrates how the sloping bed of a turning center permits the chips to fall away, while Figure 2.21b shows a horizontal machining center with the same facility. An added refinement is to have the chips continuously removed by conveyor.

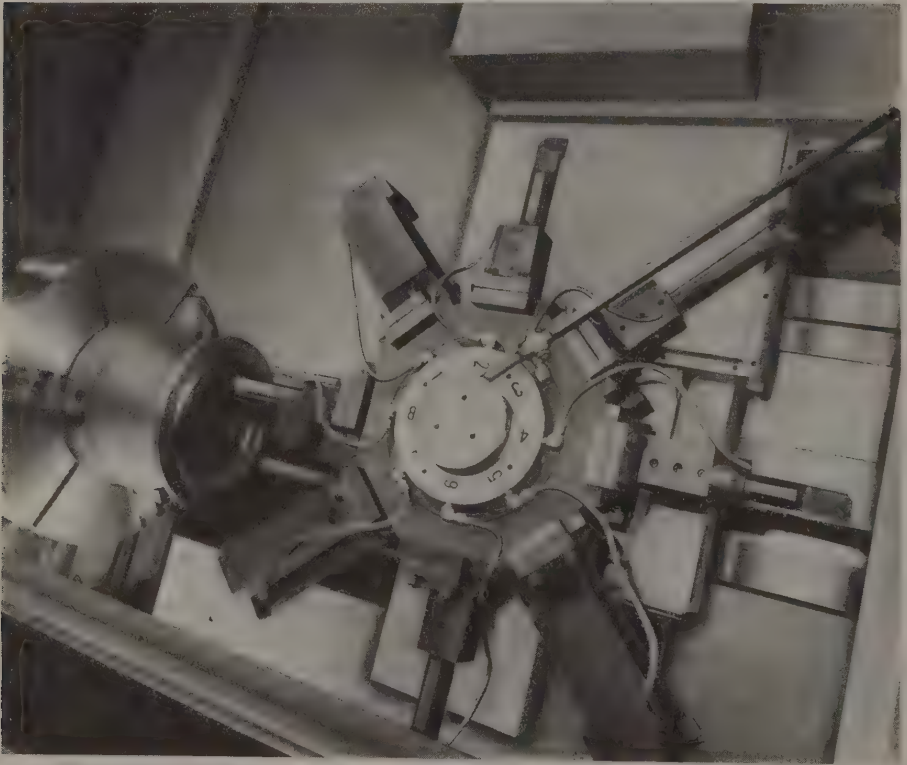


Figure 2.20 *Individual coolant supply to cutting tools.*

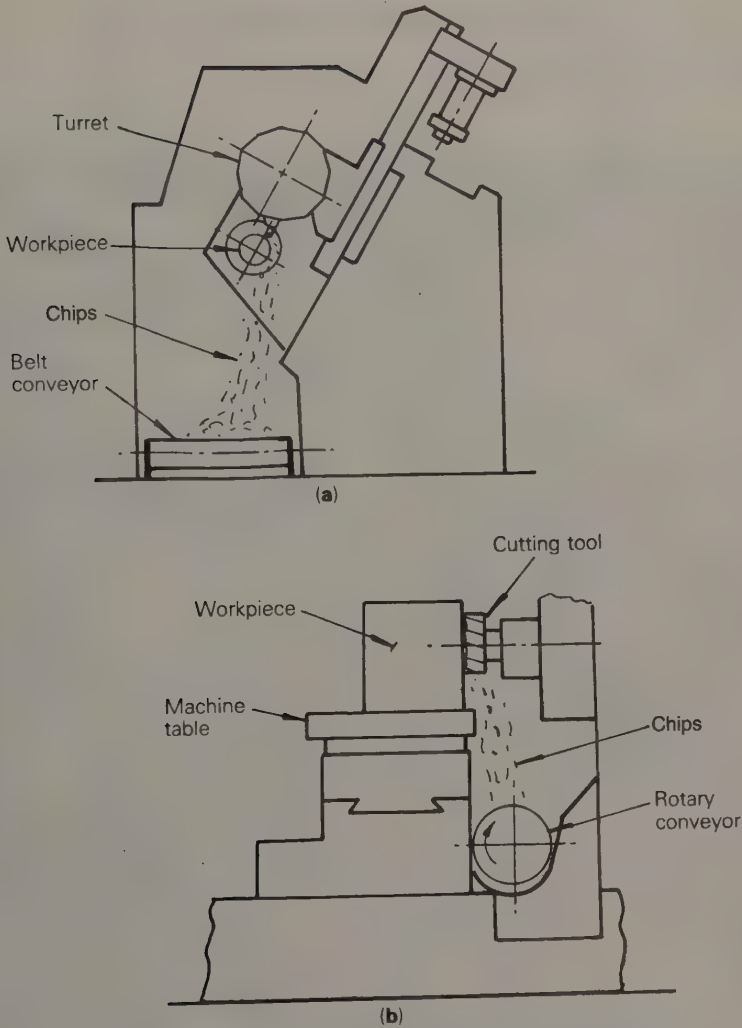


Figure 2.21 Chip removal arrangements: (a) turning center; (b) horizontal machining center.

It is best if the room temperatures for workshops containing computer numerically controlled machines are maintained at a constant 20°C or 72°F , but with recent control upgrades this is no longer mandatory. The presence of a radiator or a constantly opening and closing door close to a machine can have a detrimental effect, and very warm summer days have been known to halt production in factories where the air conditioning was inadequate.

It is also worth noting that excesses in temperature not only affect the dimensional stability of a machine but can also cause malfunctioning of the electronic control systems.



Figure 2.22 Additional machining facilities provided by second turret.

ADDITIONAL MACHINING FACILITIES

A number of machines currently available have special design features that extend their capabilities by providing machining facilities not generally available. They include the following.

1. Turning centers with two turrets positioned in such a way that two tools can cut simultaneously. An example is shown in Figure 2.22.
2. Turning centers which use special tool holders that are power driven and can be programmed to rotate when the machine spindle is stationary, thus permitting the milling of flats, keyways, and slots and the drilling of holes offset from the machine axis as illustrated in Figure 2.23.
3. A turning center with a similar facility to that described above, but where the rotating holders are located in a separate turret. In addition the spindle, when clamped to prevent rotation, can be caused to slide in the Y axis, thus giving four-axis control: X , Y , Z , and C (rotary).
4. A milling machine with two spindles providing for machining operations in the vertical and horizontal planes, a facility that is particularly useful and time saving when machining large components that cannot be readily reset. This feature is illustrated in Figure 2.24.
5. A turning center using a tooling magazine, as opposed to a turret, to provide for a greatly extended tooling range.

SAFETY

The safety aspects of computer numerically controlled machines have to be related not only to the machine operator but also to the very costly equipment. There are two problem areas: the high voltage involved and the extreme me-

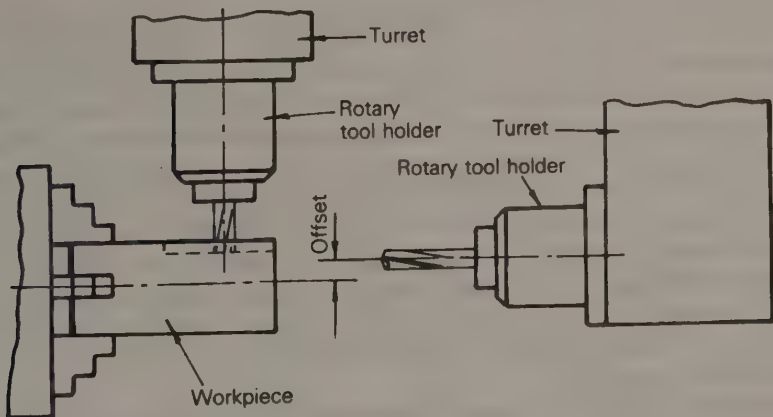


Figure 2.23 Example of additional machining facilities on a turning center.

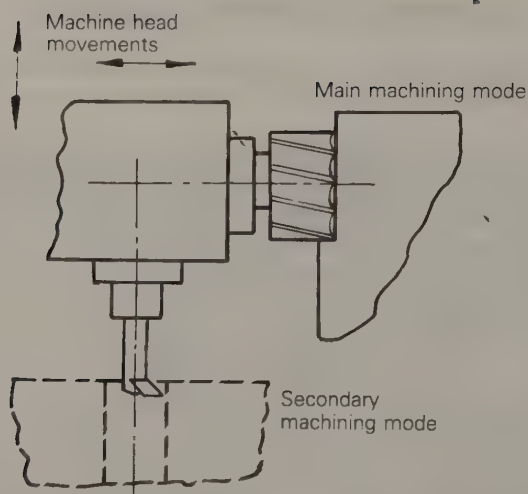


Figure 2.24 Example of additional machining facilities on a horizontal machining center.

chanical forces resulting from high spindle speeds and rapid slide movements.

The electrical services are protected by lockable covers, and access should be limited to authorized persons.

The mechanical dangers are greatly reduced by total enclosure, a common feature of computer numerically controlled machines. This affords protection from flying chips and broken tooling, reduces noise level and prevents contamination of the atmosphere by coolant, the latter being a considerable problem when spray cooling is employed.

When machines are not totally enclosed guards are used, these being fitted with interlocking switches so that there is no machine movement until the guard is correctly positioned.

The electrical control of mechanical features is also extended to work-holding devices. If the work is not correctly held, there is no machine movement.

Excessive slide movement, which could damage the machine or workpiece, is prevented by limit switches, and this can also be extended further by programmable safety zones that fall into three categories: safe, warning, and fault. Tool movement in the safe zone is unrestricted; in the warning zone it is only possible by the operator making a conscious response to a power cut-off; in the fault zone no movement at all is possible. The whole machine can, of course, be instantly immobilized by activating the obligatory emergency switch.

Reference was made earlier to conventional machines being retrofitted with computer numerical control systems. These machines often do not have the built in safety features referred to above and because of this the utmost care must be taken in their use.

The reader should appreciate that, on all computer numerically controlled machines, whether purpose built or retrofitted, slide movements can be very

rapid, and movements made at the wrong time or in the wrong direction can have disastrous results both for the operator and for the equipment. Accidents of this nature are more likely to happen during machine setting and program proving, and when restarting after a program stop.

A clearly defined code of operation, with the accent on safety, is desirable for all remote controlled machine tools, particularly those used in educational and training establishments, and the student should be made fully aware of the inherent dangers.

QUESTIONS

- 1 Define repeatability as applied to CNC machine tools.
- 2 List five qualities which need to be incorporated in a machine design if repeatability is to be maintained.
- 3 What are bifurcated structures and why are they used?
- 4 Why is it desirable that slide leadscrews should be centrally positioned?
- 5 What do you understand by the term dimensional stability as applied to CNC machine tools?
- 6 List four factors that can affect the dimensional stability of a CNC machine tool.
- 7 Explain the difference between static and dynamic loading by quoting examples of where they occur in a CNC machine tool.
- 8 Why do some manufacturers recommend a warm up period before a CNC machine tool is used?
- 9 How do some machine designs eliminate the problems caused by the heat present in chips?
- 10 List the various techniques that are used on machine-tool slides to reduce the frictional resistance to motion.
- 11 List the advantages of using DC servo motors for slide movement on CNC machines.
- 12 Why is it that gearboxes are not necessary when machine spindles are driven by DC motors?
- 13 Why is the traditional Acme form of leadscrew unsuitable for CNC machines?
- 14 List the advantages of using recirculating ball screws to facilitate slide movement.

- 15 What is the name given to the thread form of ball screw?
- 16 What is a positional transducer and where and why it is used on a CNC machine?
- 17 What are the two problem areas that make safety an important consideration in the design of CNC machine tools?
- 18 Many CNC machines are totally enclosed. List three advantages of such an arrangement.
- 19 How can the danger of inadequate work holding be eliminated?
- 20 Explain the differences among safe, warning, and fault safety zones and their purpose in CNC machine tool control systems.

3

TOOLING FOR COMPUTER NUMERICALLY CONTROLLED MACHINING

To the onlooker one of the most startling aspects of computer numerically controlled machining is the rapid metal-removal rates used. That there are cutting tools capable of withstanding such treatment can seem quite incredible. Add to this indexing times of less than one second and automatic tool changing providing a "chip-to-chip" time of around five seconds and it is easy to understand why many production engineers consider tooling to be the most fascinating aspect of computer numerically controlled machining.

MATERIALS FOR CUTTING TOOLS

Although high-speed steel (HSS) is used for small-diameter drills, taps, reamers, end mills, and spot drills, the bulk of tooling for computer numerically controlled machining involves the use of cemented carbide.

The physical properties necessary in a cutting tool are hardness at the metal-cutting temperature, which can be as high as 600°C , and toughness. High-speed steel is tougher than cemented carbide but not as hard and, therefore, cannot be used at such high rates of metal removal. On the other hand, the lack of toughness of cemented carbide presents problems, and this has meant that a tremendous amount of research has gone into developing carbide grades that, when adequately supported, are able to meet the requirements of modern machining techniques. It is only necessary to observe a computer numerically controlled machine in action to see how successful this research has been.

The hardness of cemented carbide is almost equal to that of diamond. It derives this hardness from its main constituent, tungsten carbide. In its pure form tungsten carbide is too brittle to be used as a cutting tool, so it is pulverised and mixed with cobalt.

The mixture of tungsten carbide and cobalt powder is pressed into the required shape and then sintered. The cobalt melts and binds the tungsten carbide grains into a dense, nonporous structure.

In addition to tungsten carbide, other hard materials such as titanium and tantalum carbides are used, and by providing tungsten carbide tools with a thin

layer of titanium carbide, resistance to wear and useful life are increased by up to five times.

THE PRACTICAL APPLICATION OF CEMENTED CARBIDES

Solid Tools

Solid carbide tools are somewhat restricted in their use owing to their lack of toughness. However, they are particularly useful when the work material is difficult to machine with high-speed steel, thus precluding the use of this material even for the small sizes referred to earlier. Solid carbide milling cutters as small as 1/32 in. diameter, drills as small as 1/64 in. or No. 80 (0.0135 in.) diameter and reamers as small as 1/16 in. diameter are available. The successful application of solid carbide tooling depends greatly on the tool being short and mounted with the minimum of overhang, and the machines on which they are used being vibration free and having no play or misalignment. The correct speeds and feeds have to be determined with great care, often by experiment on the particular work in hand.

Brazed Tips

When the shank size of a cutting tool is large enough, a more viable technique is to braze the carbide tip to a medium carbon steel shank. Drills, reamers, milling cutters, and turning tools produced in this way are available.

Indexable Inserts

While both types of tooling previously referred to have their particular uses, by far the most widely used application of cemented carbides is as inserts located in special holders or cartridges.

The advantages of inserts are as follows:

1. correct cutting geometry
2. precise dimensions
3. no resharpening
4. rapid replacement

The first two factors are particularly relevant where preset tooling is concerned. (Preset tooling is discussed subsequently in the text.)

The inserts are indexable, that is, as a cutting edge becomes dull the insert is moved to a new position to present a new edge to the work. The number of cutting edges available depends on the design of the insert.

The control of chips is an essential requirement when high metal-removal rates are involved, and this can be a built-in feature of the insert itself, in the

form of a groove, or of the holder, in the form of a chip-breaking pad clamped on the top of the insert.

AMERICAN NATIONAL STANDARDS INSTITUTE AND INTERNATIONAL STANDARDS ORGANIZATION CODES

Although there is still a wide range of cemented carbide grades, insert shapes, and tool-holder designs currently available, the initially somewhat confusing situation was greatly helped by the introduction of the American National Standards Institute (ANSI) and International Standards Organization (ISO) codes. Manufacturers' literature usually states where their particular products correspond with the ANSI/ISO recommendations.

Carbide Grades

Carbide grades vary according to their wear resistance and toughness. As the wear resistance increases, the toughness decreases. The ANSI/ISO systems differ in how they handle grade classification, but suppliers of carbide can provide either set of information. The ANSI system allows carbide manufacturers to create their own grade coding system, which they in turn explain in their catalog. (See example in Figure 3.1a.) The ISO code groups carbides according to their application, and they are designated by the letters P, M, and K and a number. A corresponding color code of blue (P), yellow (M), and red (K) is also used. An interpretation of the code is given in Figure 3.1b.

Inserts

Inserts are designated according to shape, size, geometry, cutting direction, etc. An interpretation of the ANSI/ISO codes is given in Figures 3.2a and 3.2b. Note that only slight differences in the two systems exist, one being the use of English or metric dimensions.

Holders and Cartridges

Holders and cartridges are designated according to a number of factors, which include tool style, method of holding the insert, tool height, and width.

The shanks of the holders and cartridges can be "qualified," that is, when the insert is located, the distance from the tool tip to a stated location face is guaranteed within a tolerance of ± 0.003 to ± 0.005 in. depending on shank size. Qualified tooling and its application are dealt with in more detail later in this chapter.

An interpretation of the ANSI/ISO codes for tool holders and cartridges is given in Figures 3.3a, 3.3b, and 3.3c. Note again that only minor differences appear between ANSI and ISO systems.

Kennametal grade system

coated grades	typical machining applications	grade composition
KC950	The carbide grade that bridges the gap from high-speed finishing to high-speed roughing. KC950 has the strength to handle interrupted cuts, even at ceramic coated insert speeds. Carbon steels, alloy steels, tool steels, ferritic and martensitic stainless steels and all cast irons.	TiC/Al ₂ O ₃ /TiN coatings on a tough, heat resistant carbide substrate
KC910	Excellent abrasion resistance for long tool life in high-speed finishing to light roughing operations. Carbon steels, alloy steels, tool steels, ferritic and martensitic stainless steels and all cast irons.	TiC/Al ₂ O ₃ coatings on a very thermal-deformation-resistant carbide substrate
KC850	The toughest coated grade for your toughest jobs. Finishing to heavy roughing, depending on insert geometry. Unsurpassed thermal and mechanical shock resistance makes KC850 ideally suited for applications requiring maximum edge strength. Carbon steels, alloy steels, tool steels, austenitic stainless steels, alloy cast iron, ductile iron.	TiC/TiC-N/TiN coatings on a specially strengthened carbide substrate
KC810	Reliable performance from finishing to moderate roughing at moderate speeds. Good balance of wear resistance and strength for general purpose machining. Carbon steels, alloy steels, and tool steels.	TiC/TiC-N/TiN coatings on a carbide substrate
KC250	Light to heavy roughing of stainless steels, high temperature alloys and cast irons. Excellent mechanical shock resistance at low to moderate speeds.	TiC/TiC-N/TiN coatings on a very tough carbide substrate
KC210	A supplementary grade for finishing to light roughing of cast irons, stainless steels and some high temperature alloys. Excellent wear resistance at moderate speeds.	TiC/TiC-N/TiN coatings on a carbide substrate
uncoated steel cutting grades	typical machining applications	physical characteristics
K45	Primary uncoated grade for finishing and light roughing of all steels. Excellent crater, and edge wear resistance. Frequently applied in grooving where maximum edge wear resistance is required.	↑ increasing hardness ↓ increasing toughness
K4H	Excellent for threading steels and cast irons. Also may be used for semi-finishing to light roughing of steels and cast irons at moderate speeds and moderate chip loads.	
K2884	General purpose steel milling grade that may be used in moderate to heavy chip loads. Excellent edge wear and mechanical shock resistance	
K2S	A supplementary grade for light to moderate roughing of steels at moderate speeds and feeds	
K430	Primary uncoated grade for heavy roughing to semi-finishing of all steels. Superior edge strength and thermal shock resistance for milling or turning through severe interruptions at high chip loads.	
K21	A supplementary grade for heavy to light roughing of steels at low to moderate speeds. Good mechanical and thermal shock resistance.	
uncoated cast iron grades	typical machining applications	physical characteristics
K46	Primary uncoated grade for machining stainless steels, cast irons, non-ferrous alloys, nonmetals, and most high temperature alloys. Excellent edge wear resistance.	↑ increasing hardness ↓ increasing toughness
K6	Moderate roughing grade for cast irons, nonferrous alloys, nonmetals and most high temperature alloys. High edge strength and good wear resistance.	
K0735	Excellent milling grade for gray, malleable and nodular cast irons at high speeds and light chip loads. Superior resistance to built-up-edge in machining all stainless steels and aluminum alloys.	
K1	Excellent mechanical shock resistance for roughing through heavy interruptions when turning or milling stainless steels, most high temperature alloys including titanium, cast irons and cast steels, and rough cast nonferrous alloys.	

TiC titanium carbide
TiN titanium nitride

TiC-N titanium carbo-nitride
Al₂O₃ aluminum oxide

(1) Primary grades—recommended for most machining applications

Kennametal/85 Catalog Copyright 1984. Kennametal Inc., Advertising Dept.,
P.O. Box 231, Latrobe, PA 15650-9984.

Figure 3.1a Manufacturer's ANSI carbide insert grade classification system. (Note: Different manufacturer's use different coding systems.)

Kennametal advanced cutting materials

cutting material	typical machining applications	material composition
Kyon 2000	The first true, high velocity, roughing grade for nickel base alloys and cast iron. See page 294 for additional information.	Sialon: silicon nitride and aluminum oxide
K090	High velocity finishing and semi-finishing at 2 to 3 times the speed of carbides. Cast irons, alloy steels over 330 BHN and nickel base alloys over 260 BHN.	composite ceramic: aluminum oxide and titanium carbide.
K080	Finish machining of cast iron and steels below 330 BHN. Excellent edge wear resistance.	high purity, cold pressed aluminum oxide.
KD100	The answer to high velocity, high volume production machining of abrasive non-ferrous materials. High silicon aluminum, fiber reinforced plastics, phenolics, etc. See page 295 for additional information.	Synthetic polycrystalline diamond compact
KD120	A problem solver for close tolerance finishing of hardened ferrous materials. Alloy steels and cast irons over 450 BHN. See page 295 for additional information.	Polycrystalline cubic boron nitride compact

Figure 3.1a (Continued)

Figure 3.4 shows the four recommended methods of locating and clamping inserts in holders and cartridges.

THE PRACTICAL APPLICATION OF INDEXABLE CARBIDE INSERTS

Figure 3.5 shows how a variety of insert shapes may be applied to produce external and internal turned profiles, while Figure 3.6 shows a range of holders and inserts. Figures 3.7 and 3.8 show applications of inserts to milling operations. Figures 3.9a and 3.9b show the application of cartridges to boring while Figure 3.9c shows their application to a face milling cutter body.

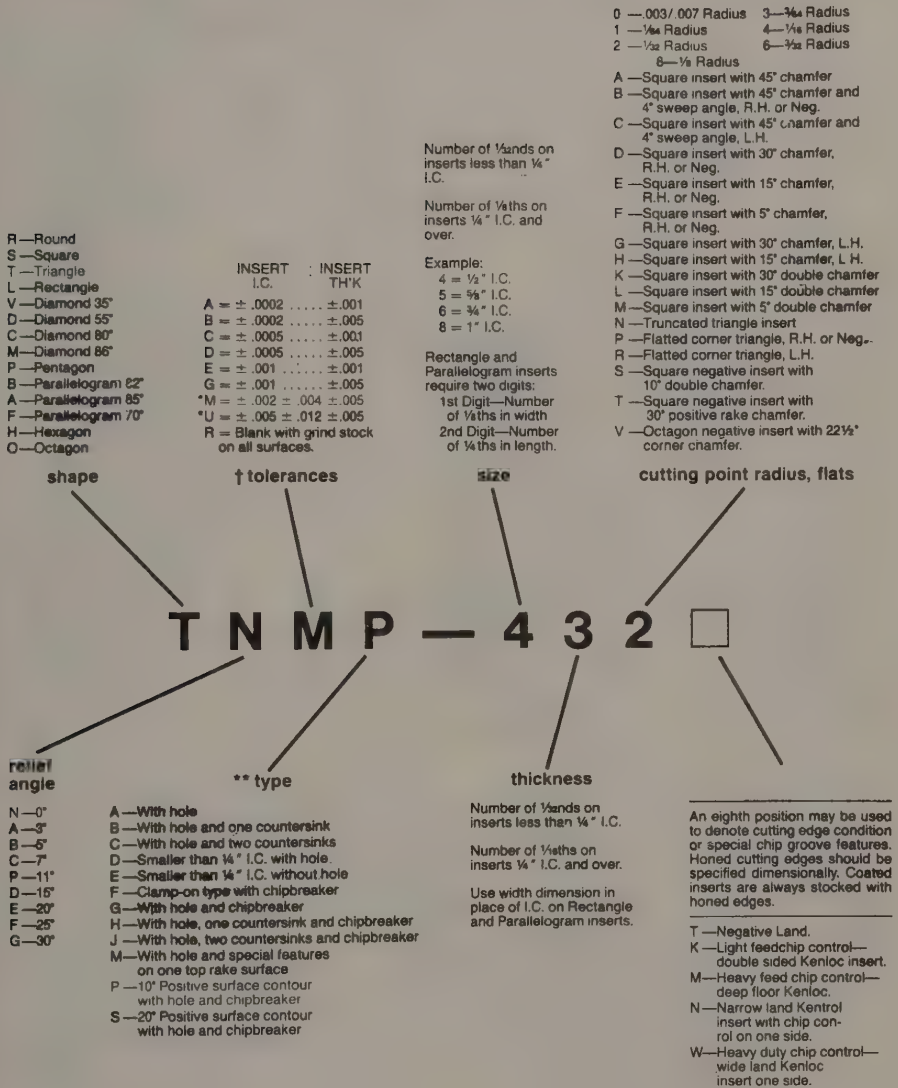
METALCUTTING SAFETY

Modern metalcutting operations involve high energy, high spindle or cutter speeds, and high temperatures and cutting forces. Hot, flying chips may be projected from the workpiece during metalcutting. Although the nonductile cemented carbide and ceramic cutting tools used in metalcutting operations are designed and manufactured to withstand the high cutting forces and temperatures that normally occur in these operations, they are susceptible to fragmenting in service, particularly if they are subjected to over-stress or severe impact, or are otherwise abused. Therefore, precautions should be taken to protect adequately workmen, observers, and equipment against hot, flying chips, fragmented cutting tools, broken workpieces, carbide particles, or other similar projectiles. Machines should be fully guarded and personal protective equipment should be used at all times.

ISO Code		Application
Color code: Blue	P01	For finishing steel, high cutting speeds, light feeds, favorable conditions.
	P10	Slightly tougher grade for finishing and light roughing steel and castings. No coolant.
	P20	For medium roughing of steel, less favorable conditions. Moderate cutting speeds and feeds.
	P30	For general-purpose turning of steel and castings, medium roughing.
	P40	For heavy roughing of steel and castings. Intermittent cutting, low speeds and feeds
	P50	For difficult conditions. Heavy roughing intermittent cutting. Low cutting speed and feed.
Color code: Yellow	M10	For finishing stainless steel using high cutting speeds.
	M20	For finishing and medium roughing of alloy steels.
	M30	For light to heavy roughing of stainless steel and materials difficult to cut.
	M40	For roughing tough-skinned materials using low cutting speeds.
Color code: Red	K01	For finishing plastics and cast iron.
	K10	For finishing brass and bronze using high cutting speeds and feeds.
	K20	For roughing cast iron. Intermittent cutting, low speeds, high feeds.
	K30	For roughing and finishing cast iron and non-ferrous materials. Favourable conditions.

Figure 3.1b Selection of ISO carbide grades for metal-cutting applications.

indexable inserts identification system



*Exact tolerance is determined by size of insert

**Shall be used only when required

†A & B I.C. Tolerances only in uncoated grades

Kennametal/85 Catalog Copyright 1984. Kennametal Inc., Advertising Dept.,

P.O. Box 231, Latrobe, PA 15650-9984

Figure 3.2a ANSI indexable inserts identification system.

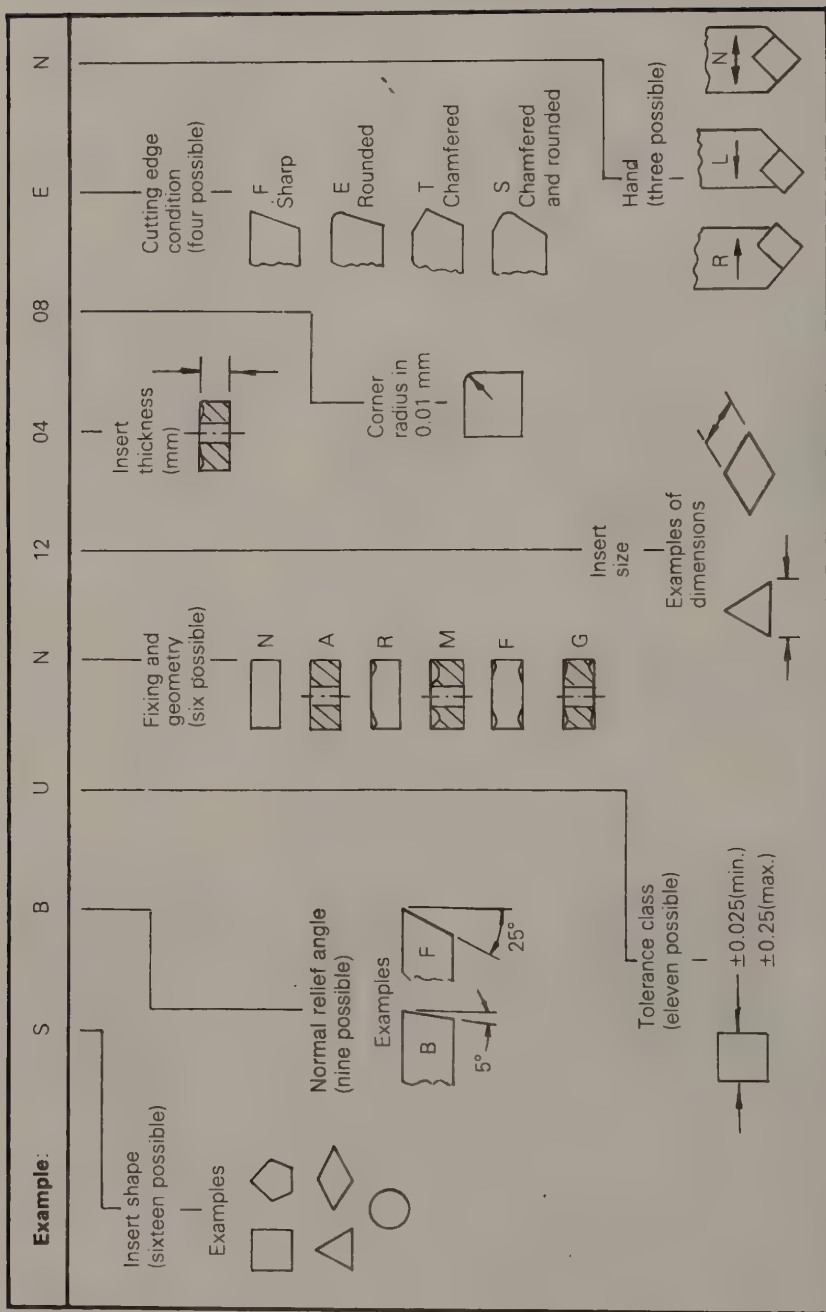


Figure 3.2b Interpretation of ISO 1832: 1977 designation of indexable inserts.

toolholder identification system

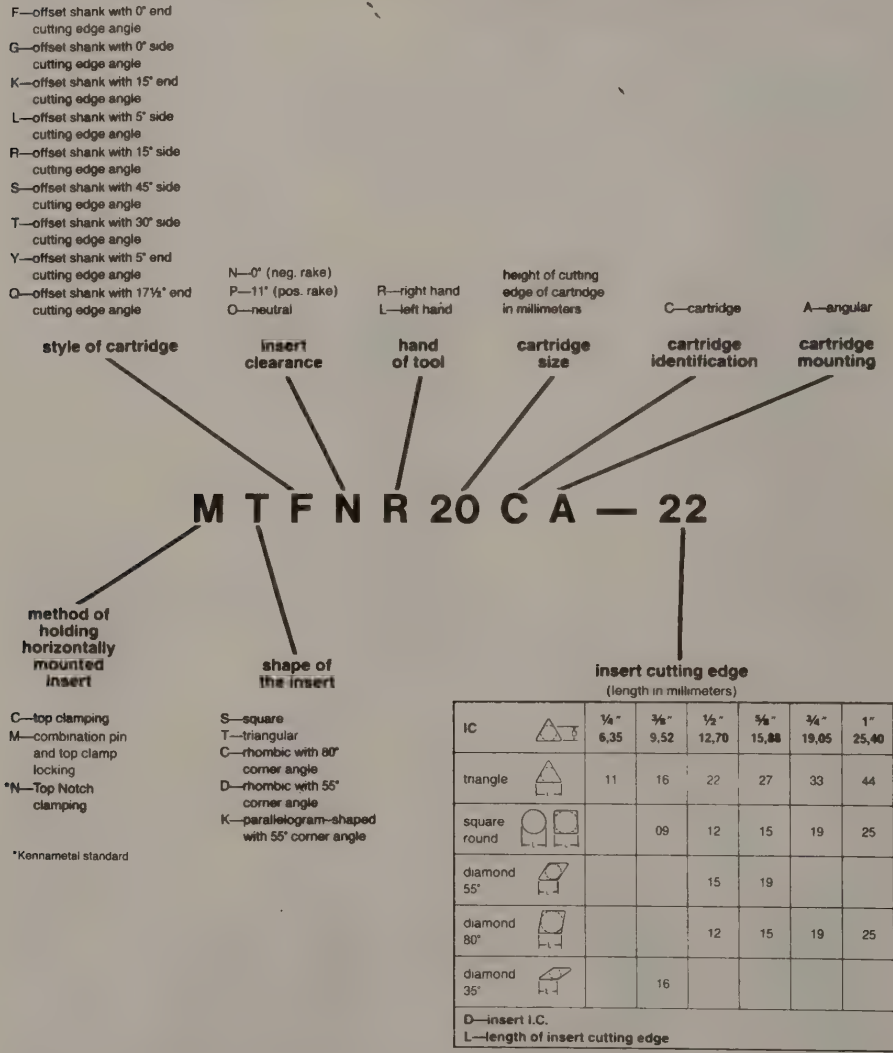
This identification system was developed for qualified holders, and has been used in listing the catalog numbers for qualified holders shown in this catalog.



Kennametal/85 Catalog Copyright 1984. Kennametal Inc., Advertising Dept., P.O. Box 231, Latrobe, PA 15650-9984.

Figure 3.3a ANSI toolholder identification system.

ANSI/ISO cartridge identification system



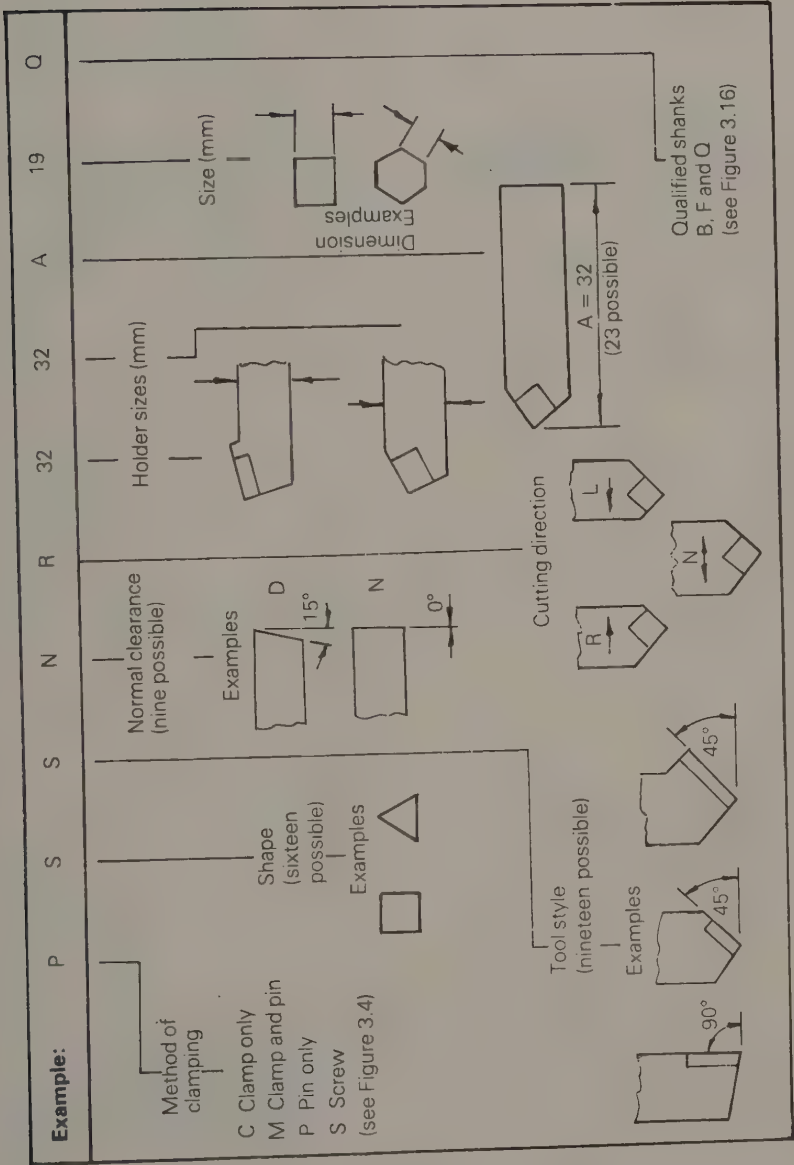


Figure 3.3c Interpretation of ISO 1832: 1977 designation of tool holders and cartridges.

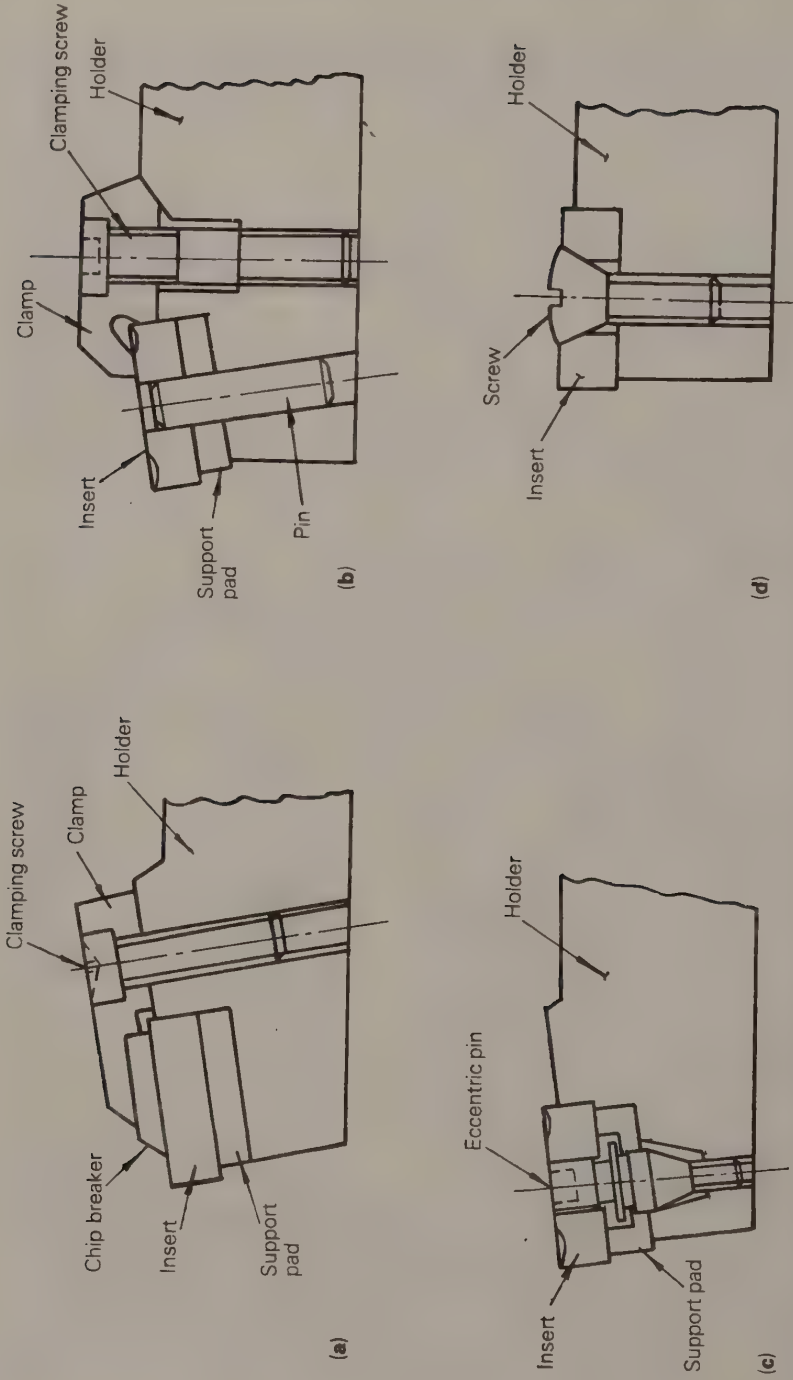


Figure 3.4 Insert clamping arrangements: (a) clamp only, (b) clamp and pin, (c) pin only, (d) screw.

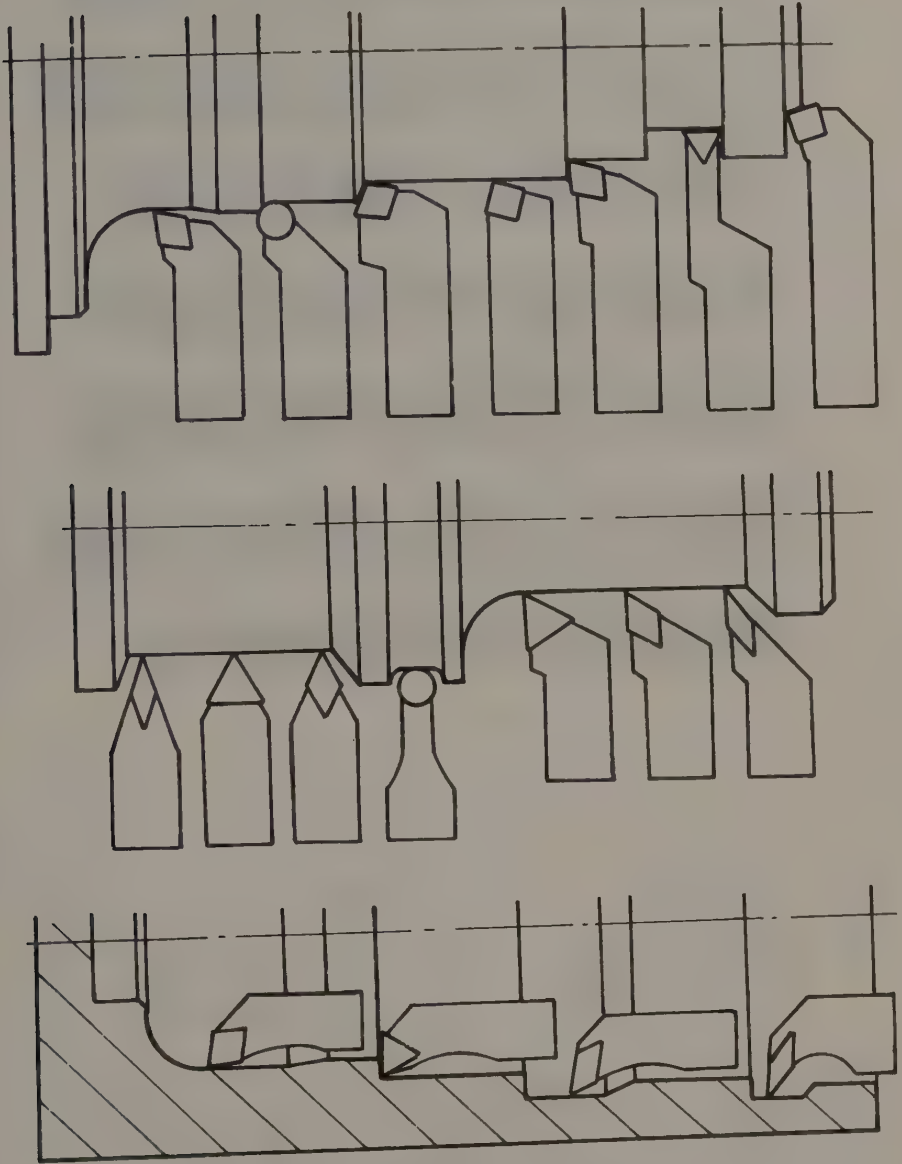


Figure 3.5 Use of insert shapes for various turning operations.

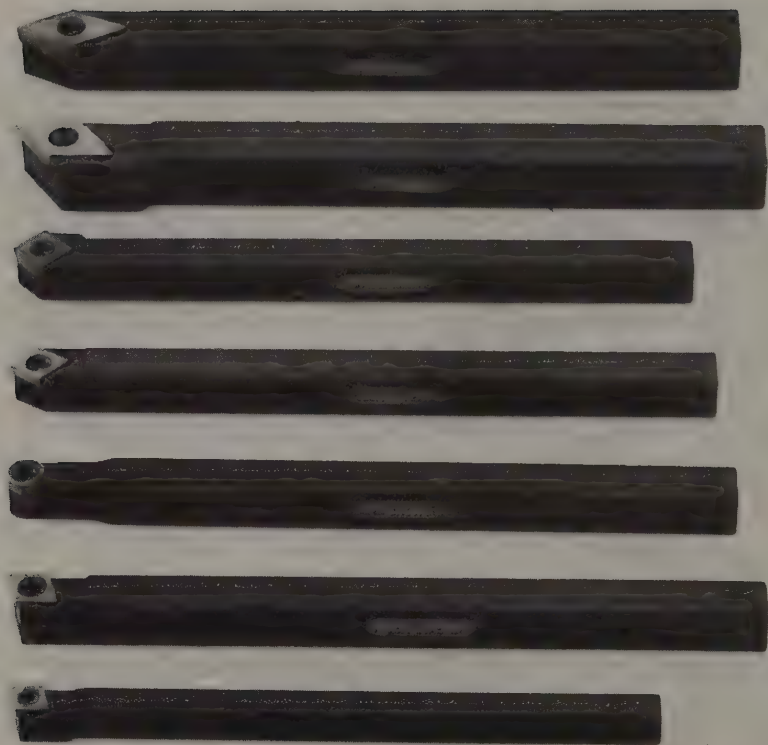


Figure 3.6 Range of indexable insert turning tools.

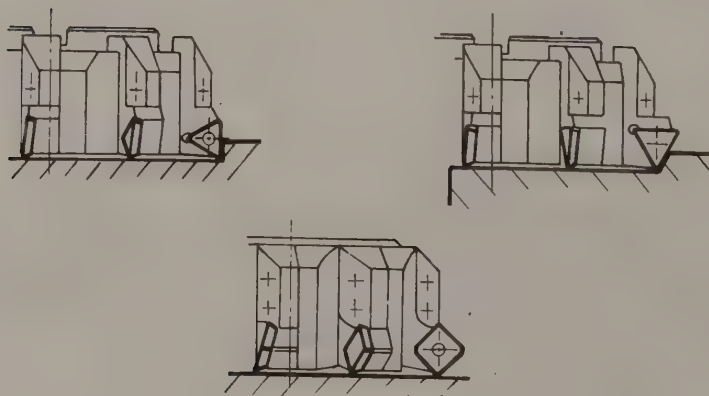


Figure 3.7 Application of various insert shapes to face milling cutters.

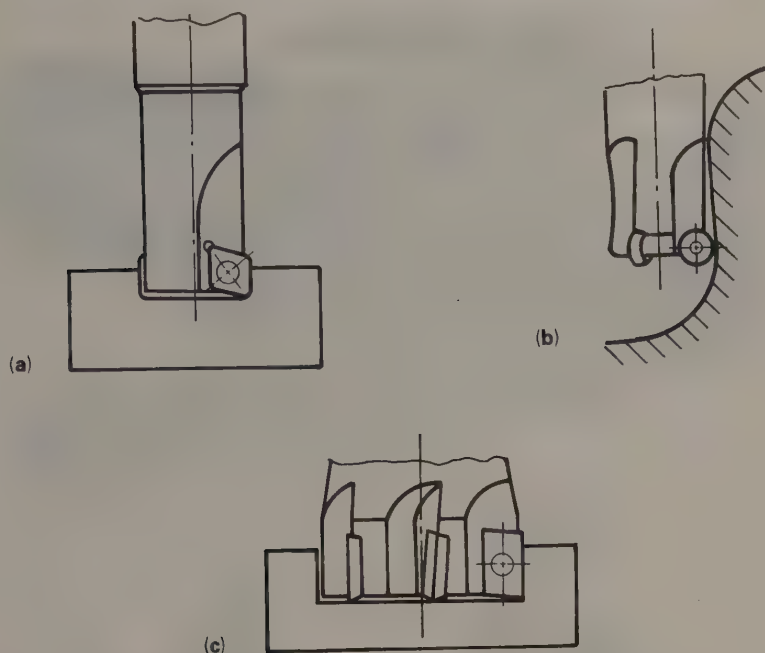


Figure 3.8 Application of insert cutters to various milling operations: (a) boring, (b) contouring, and (c) slot milling.

When grinding carbide and ceramic cutting tools, or otherwise fabricating cemented carbide, a suitable means for collection and disposal of dust, mist, or sludge should be provided. Inhalation of dust or mist containing metallic particles can be hazardous, particularly if exposure continues over an extended period of time. Therefore, adequate ventilation should be provided. General Industry Safety and Health Regulations, Part 1910, U.S. Department of Labor, published in Title 29 of Federal Regulations, particularly those sections dealing with ventilation, local exhaust systems, and occupational health and environmental control as it relates to cobalt (Co), metal fume and dust, and tungsten (W), as well as other government regulations, should be consulted.

Tungsten carbide and ceramic cutting edges, and related supporting holders such as milling cutters and boring bars, are only one part of the man-machine-tool system. Many variables exist in machining operations, including the metal removal rate; the workpiece size, shape, strength, and rigidity; the chucking or fixturing; the load carrying capability of centers; the cutter and spindle speed and torque limitations; the holder and boring bar overhang; the available power; and the condition of the tooling and the machine. A safe metalcutting operation must take all of these variables, and others, into consideration.



Kennametal/85, Copyright 1984. Kennametal Inc., Advertising Dept.,
P.O. Box 231, Latrobe, PA 15650-9984.

Figure 3.9a ANSI/ISO cartridges in boring bar applications.



Kennametal/85 Copyright 1984. Kennametal Inc., Advertising Dept.,
P.O. Box 231, Latrobe, PA 15650-9984.

Figure 3.9b *Tenthset precision boring cartridge with front adjustment of 0.0001 in.*



Figure 3.9c Applications of cartridges to a face milling cutter.

TOOLING SYSTEMS

The production of a machined component invariably involves the use of a variety of cutting tools, and the machine has to cater for their use. The way in which a range of cutting tools can be located and securely held in position is referred to as a tooling system and is usually an important feature of the machine tool manufacturers' advertising literature.

The tooling system for a machining center is illustrated in Figure 3.10. Note the use of tool holders with standard tapers, a feature that can be very helpful in keeping tooling costs to a minimum.

The types of tool holders shown in Figure 3.10 are retained in and released from the machine spindle by a hydraulic device, an arrangement that lends itself to automation, since it is relatively simple to control hydraulic systems using electrically activated solenoid valves which themselves can be controlled via the machine control system. The hydraulic force retaining the holder is supplemented by a mechanical force exerted by powerful disk springs, as illustrated in Figure 3.11, for added safety. It should be noted that less expensive machines may use a mechanical device in conjunction with pneumatics or hydraulics for tool changes.

Not all machines have automatic tool-changing arrangements and when manual tool changing is involved, mechanical retaining devices are used. Conventional tool holders for milling situations use the tried and tested screwed drawbar arrangement, but unfortunately their use is not in keeping with modern machining techniques, where the accent is on speed. Because of this, several machine tool manufacturers have introduced tool holders of their own design that have dispensed with the need to undo a drawbar each time a tool holder is changed and as a result they have greatly speeded up the replacement process.

As with milling, a tooling system for a turning center will indicate the range of tooling which can be accommodated on the machine. One such system is illustrated in Figure 3.12.

TOOL IDENTITY

The automatic selection and presentation of a cutting tool to the workpiece is a prime function of computer numerically controlled machining. To achieve this there must be a link between programming and machine setup. Tool stations are numbered according to the tooling stations available (see Figures 2.1 and 3.13), and when writing a program, the programmer will provide each tool with a corresponding numerical identity, usually in the form of the letter T followed by two digits: T01, T02, T03 and so on. The machine setter will need to know the type of tool required and will set it in its allocated position. The transfer of information between the programmer and the machine setter is discussed in more detail in Chapter 8.

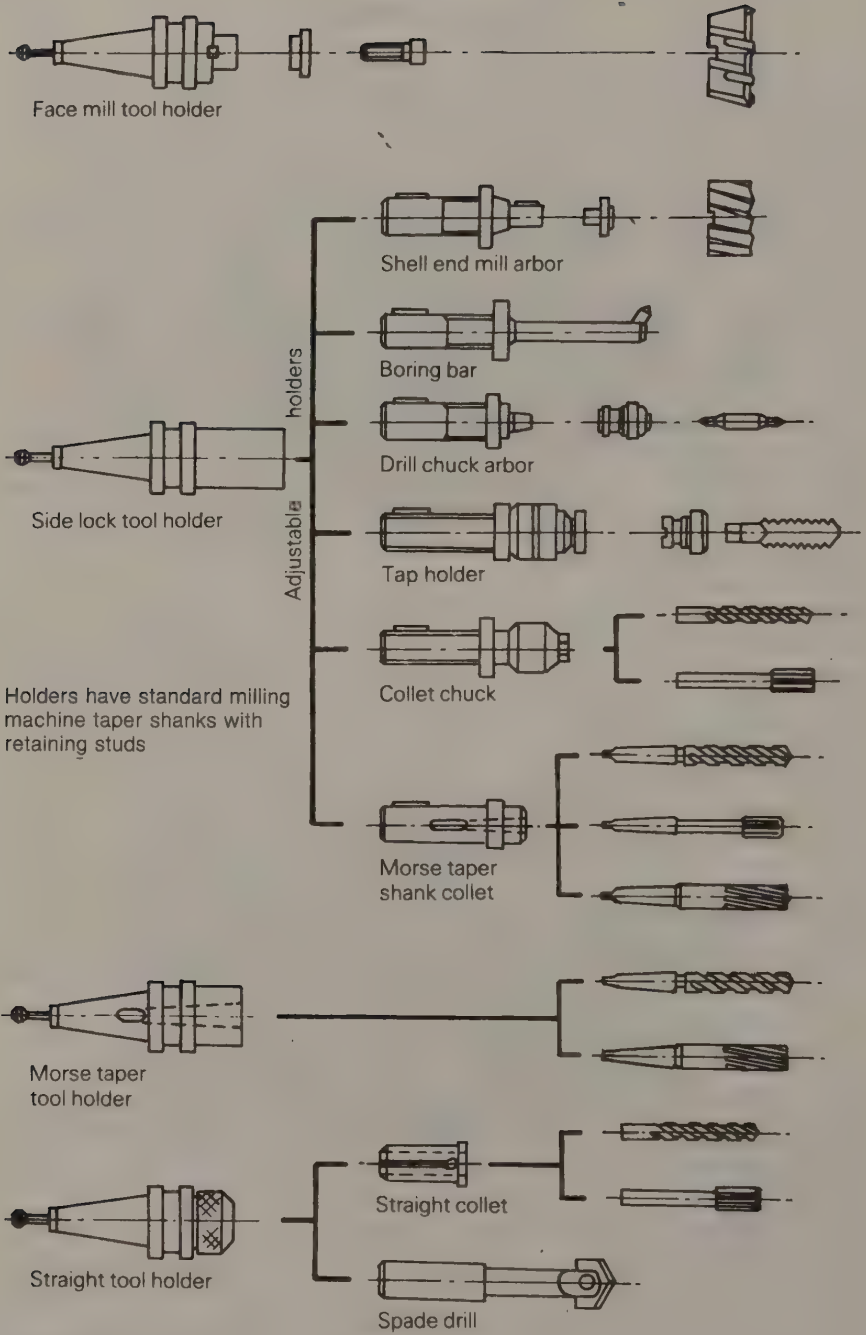


Figure 3.10 Tooling system for a machining center.

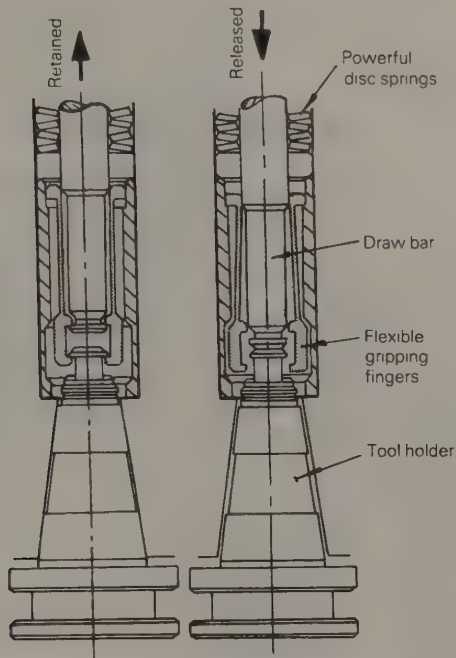


Figure 3.11 Hydraulic-mechanical draw bar assembly used on machining centers.

INDEXABLE TURRETS

The turret is the part of some machines in which the cutting tools are located. They are automatically indexable, that is, they can be programmed to rotate to a new position so that a different tool can be presented to the work. Indexable turrets are used on the majority of turning centers and on some milling/drilling machines.

There are a number of turret configurations currently available on turning centers. Several different types appear in the illustrations used throughout this book. The number of tools that can be accommodated varies with machine type, but eight or ten tool positions are usually sufficient to satisfy most machining requirements, and in many cases a standard setup consisting of a range of external and internal turning tools is advised.

There are some machines in which the turret is removable and, if two turrets are available, the spare one can be loaded with tools for a particular job before they are needed and then the turret is attached to the machine when required, a technique that reduces the machine down time considerably.

A variation of the rotating turret is the indexable slide on which the tools are mounted. The manufacturers of this particular arrangement claim that linear

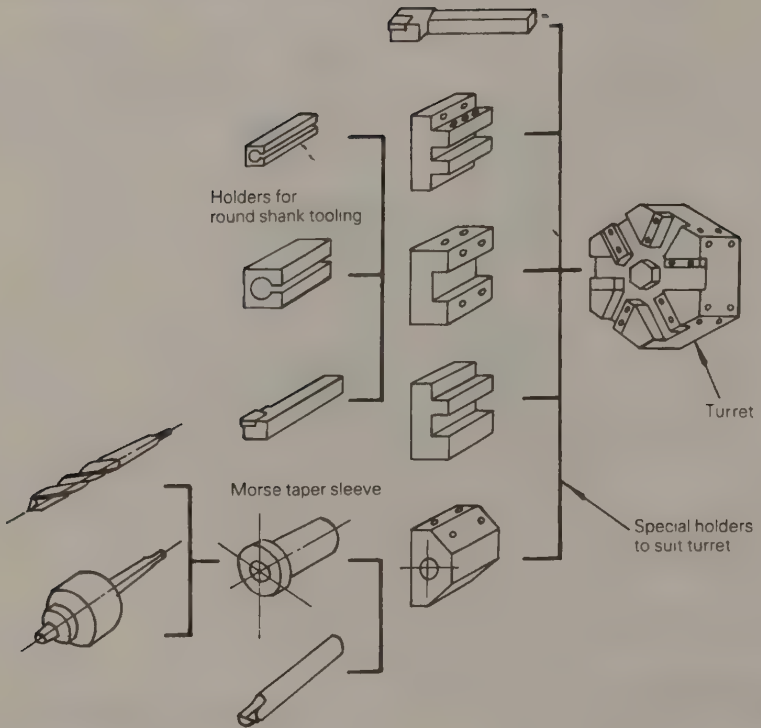


Figure 3.12 Tooling system for turning center.

tool indexing is much more rapid than rotary indexing. An optional extra available is a subbase plate to which the tools may be attached away from the machine. As with the removable turret referred to, this base plate is interchangeable and so a spare one can be loaded with tools in advance and then quickly attached to the machine when required.

Turrets generally fitted to milling/drilling machines are somewhat different from those fitted to turning centers, because each tooling position is in fact a spindle that has to rotate at a predetermined speed. Only the tool in the machining position will rotate, the others remaining stationary. A turret of this type, with ten tooling positions, is shown in Figure 3.13. (The concept of individually rotating tool holders has been extended more recently to turning centers. See Chapter 2.)

TOOL MAGAZINES

A tool magazine is an indexable storage facility used on machining centers to store tools not in use. The most common types of magazines are the rotary

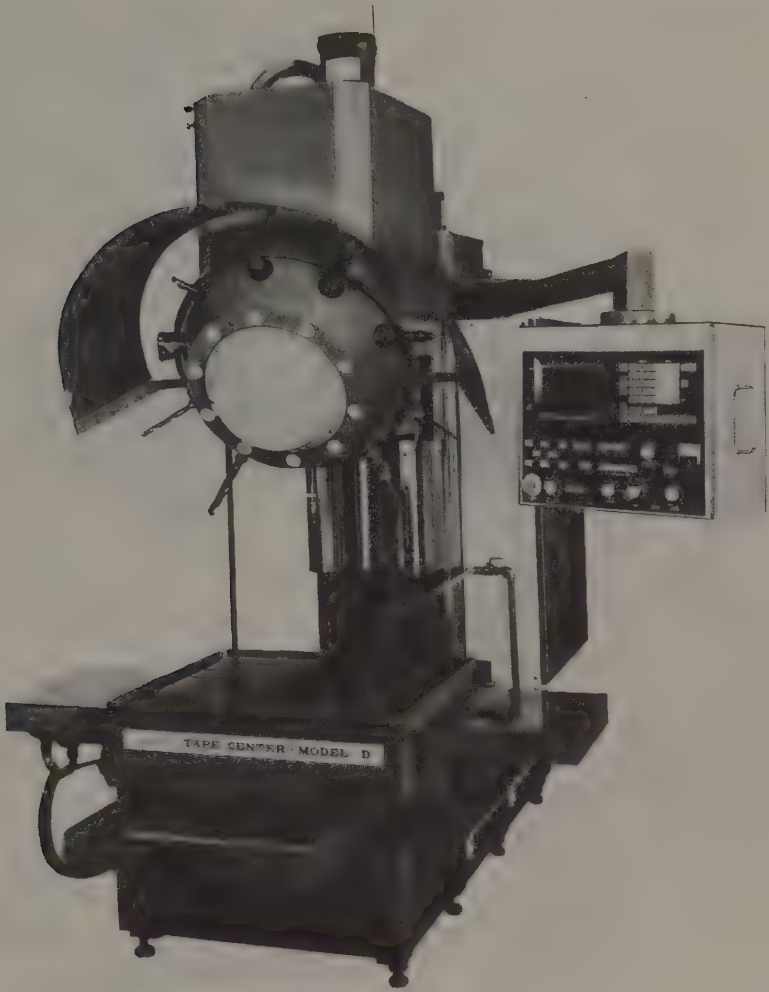


Figure 3.13 *Milling/drilling center with indexable turret.*

drum type illustrated in Figures 2.1a and 2.1b, the turret type in Figure 3.13, and the chain type illustrated in Figure 3.14. When a tool is called into use, the magazine will index, on most machines by the shortest route, to bring the tool to a position where it is accessible to a mechanical handling device. When the tool is no longer required, it is returned to its allotted position in the magazine prior to the magazine indexing to the next tool called.

The position of the tool magazine in relation to the spindle varies from one machine to another. There are also variations in the design of the tool-handling

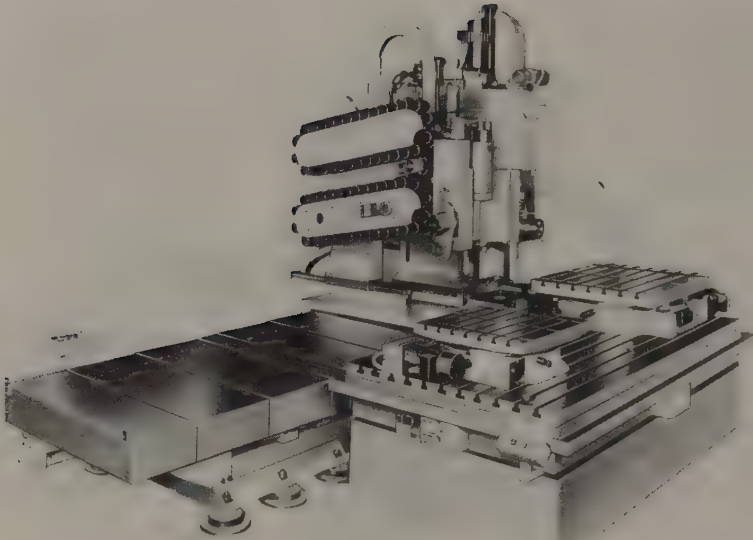


Figure 3.14 *Horizontal machining center with chain-type tooling magazines.*

devices. The two features are, of course, interrelated. Two arrangements are shown in Figure 3.15.

The capacity of magazines is another variable feature, with 12 to 24 stations being typical numbers for the rotary drum magazines and from 24 to 180 for the chain type.

REPLACEMENT TOOLING

From time to time, owing to wear or breakage, cutting tools have to be replaced. Such changes need to be rapid, with the minimum loss of machining time.

If the machining program is to remain valid, one of two requirements must be met:

1. The replacement tool must be dimensionally identical to the original.
2. The program must be capable of temporary modification to accommodate the tool variations.

Identical replacement tooling can be achieved by using qualified or preset tooling. Temporary program modifications are achieved by offsetting the tool from its original datum.

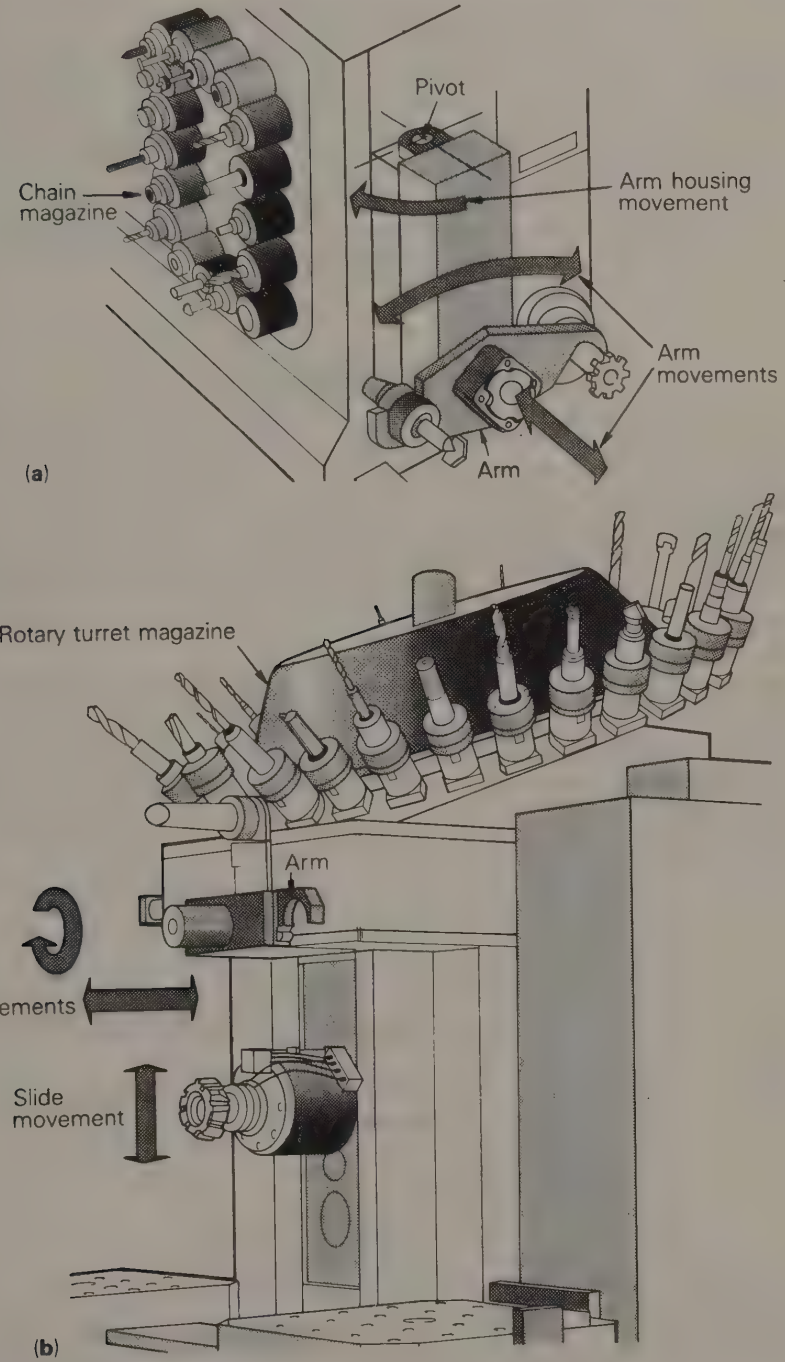


Figure 3.15 Auto tool changers.

QUALIFIED TOOLING

The ISO code illustrated in Figure 3.3c refers to qualified tooling. The dimensions from up to three datum faces to the tool tip can be guaranteed within ± 0.08 mm (± 0.003 in.). Thus if the tolerance on the dimension being machined is such that a variation in size within ± 0.08 mm (± 0.003 in.) is acceptable, one tool can readily be replaced by another. Precise location of the holder or cartridge in the machine turret or spindle is an essential feature of replacing tooling of this type. The qualified dimensions are illustrated in Figure 3.16.

PRESET TOOLING

Preset tooling involves setting the cutting edge of the tool in relation to a datum face to predetermined dimensions, these dimensions having been taken into consideration when the part program was written. A simple explanation of pre-setting is given in Figure 3.17. The through hole in the component is produced by drilling, and drill A is to be replaced by drill B. Since the depth of travel on a through hole is not precise, it would be sufficient to set the projecting length of the drill with a rule and, providing dimension X is the same for the replacement drill as it was for the original, the program would still be valid.

When closer tolerances are involved, the setting technique will need to be more precise. A wide range of specialized equipment is commercially available, the basic requirements of such equipment being a dummy tool locating/holding device with datum faces and appropriate measuring instruments. The tip of the tool is then set to predetermined dimensions in relation to datums as illustrated in Figure 3.18.

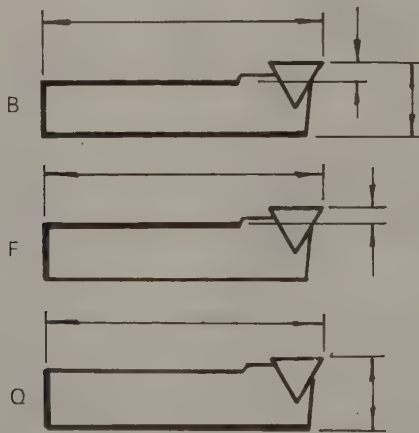


Figure 3.16 Qualified tooling. Dimensions indicated guaranteed to within ± 0.08 mm (0.003 in.).

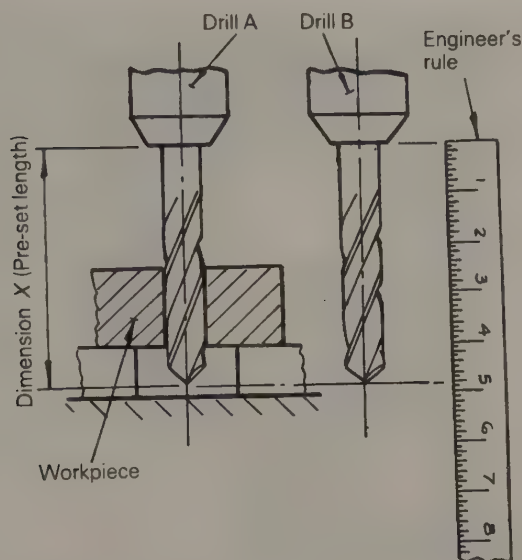


Figure 3.17 Simple presetting of tool length.

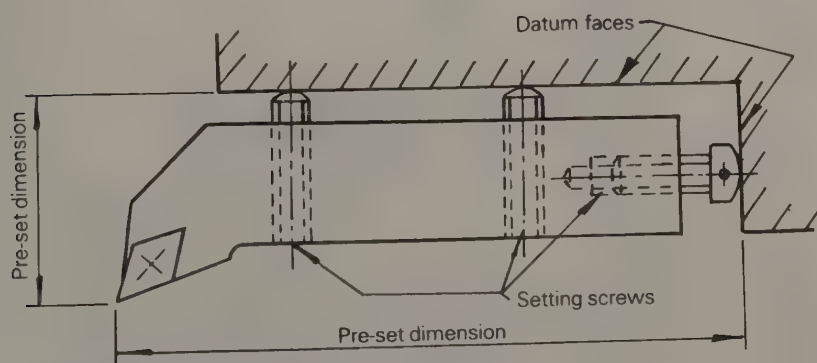


Figure 3.18 Principle of presetting a tool holder.

The more elaborate tool-setting equipment uses optical projection. An example is shown in Figure 3.19.

CUTTER COMPENSATION

The basis of computer numerical control is programming machine slide movements to occur over a stated distance in relation to a predetermined datum. Generally there is one datum for each axis of movement. However, most machining operations involve the use of more than one tool, varying in length or

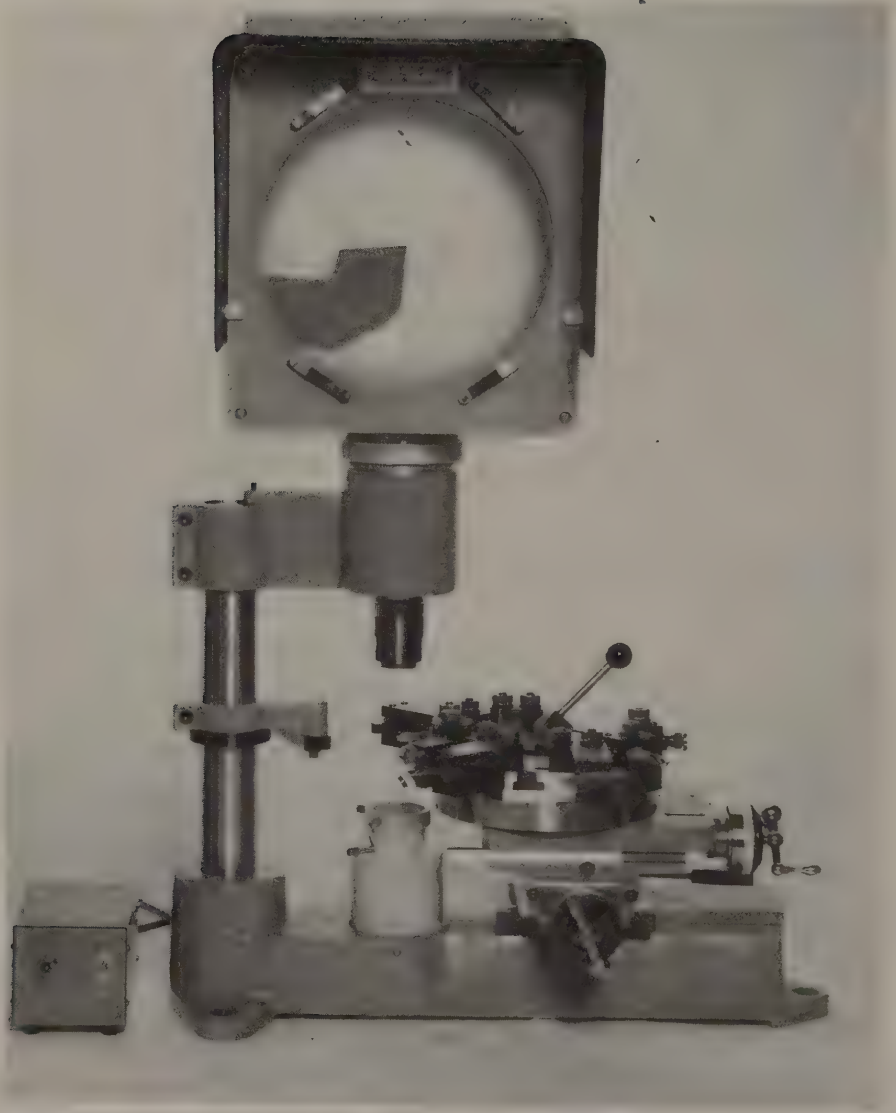


Figure 3.19 *Presetting tooling on a replacement turret using optical projection.*

diameter, which means that if the cutting edge of one tool is set to the datum to which slide movements are to be related, tools that have dimensional variations from the set tool will not start their movements from the same datum. Some compensation in slide movement is necessary to accommodate the dimensional variations of the tools. This compensation is referred to as tool offset and the offset facility is available only on computerized numerically controlled

machines. Once the offset has been established, the slide movement is automatically adjusted as required during the program run.

An offset is therefore a dimensional value defining the position of the cutting edge, or edges, of a tool in relation to a given datum.

Tool Length Offsets

Consider the component shown in Figure 3.20a. The programmer has decided that the Z datum clearance plane will be 0.1 in. (2 mm in Figure 3.20b) above the top face of the work. All tool movement in the Z axis will be in relation to that datum. The machine setter or operator will establish the datum either by “touching on” to the work surface and moving away 0.1 in. (2 mm in Figure 3.20b) or by touching on to a suitable 0.1 in. (2 mm in Figure 3.20b) thick setting block, and then setting the Z axis readout to zero.

Now consider the tooling shown in Figure 3.20b and assume the tool T01 has been set as described above. This is now the master tool. However, the machining that is required also involves using tools T02 and T03 and the position of their cutting edges in the Z axis does not correspond to that of tool T01. Tool T02 is too short and tool T03 is too long. Any movements in relation to the Z zero axis involving these tools must take into account their starting position.

The tool setter or machine operator must therefore establish the length and direction of movement which is necessary to bring the end of each tool to the

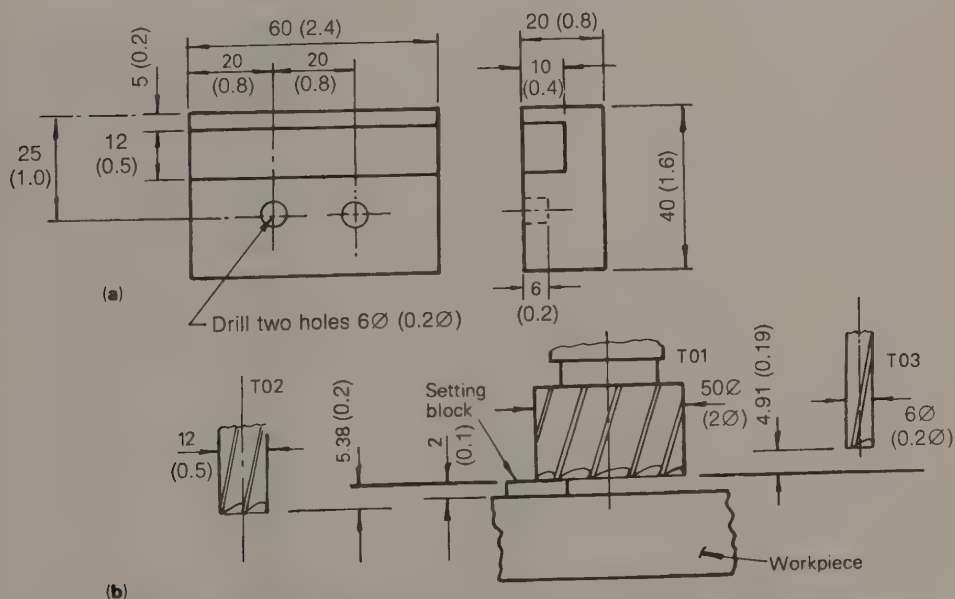


Figure 3.20 (a) Component detail and (b) tool length offsets for the milling/drilling operation. (Inch units are given in parentheses.)

zero position. This can be done by touching each tool on to the setting block and noting on the digital readout its variation from zero. Tool T02 will require a movement in the Z axis of 0.2 in. (5.38 mm) and tool T03 a movement of -0.19 in. (-4.91 mm). These dimensions are the tool length offsets.

Having established the offsets, the operator records them either by setting a series of numbered thumb-wheel switches or, as is the case with most modern control units, by entering them via the control panel key pad into an offset file or page, which can be displayed on the controller visual display screen. The method of entering offsets and the display format vary according to the control unit. The entry for the offsets relating to the tooling in Figure 3.20 could appear as shown in Figure 3.21.

With the tool length offsets being established at the machine, the part programmer is now able to ignore the variations in tool length and write the program on the assumption that all tools are starting their movements from the Z axis zero datum.

	T	Length	Diameter
	1	0.0000	2.0000
	2	-0.2000	0.5000
	3	0.1900	0.2000
	4	0.0000	0.0000
	5	0.0000	0.0000
(a)	6	0.0000	0.0000
	7	0.0000	0.0000
	8	0.0000	0.0000
	9	0.0000	0.0000
	10	0.0000	0.0000
	11	0.0000	0.0000
	12	0.0000	0.0000
	13	0.0000	0.0000
	14	0.0000	0.0000

	T	Length	Diameter
	1	0.0000	50.0000
	2	-5.3800	12.0000
	3	4.9100	6.0000
	4	0.0000	0.0000
	5	0.0000	0.0000
	6	0.0000	0.0000
(b)	7	0.0000	0.0000
	8	0.0000	0.0000
	9	0.0000	0.0000
	10	0.0000	0.0000
	11	0.0000	0.0000
	12	0.0000	0.0000
	13	0.0000	0.0000
	14	0.0000	0.0000

Figure 3.21 (a) Inch and (b) metric tool offset file for milling.

Tool length offsets are not confined to milling. They are also applicable to turning, but in this case two offset lengths are involved, one in the X axis and the other in the Z axis. The set of tools with varying lengths shown in Figure 3.22 illustrates the situation, while Figure 3.23 shows how the necessary offsets would be entered in an offset file.

Tool Radius Offsets

Just as cutting tools vary in length, they may also vary in diameter or, in the case of turning tools, in the radius of the tool tip.

Consider the profile shown in Figure 3.24. This profile could be machined by a cutter of, say, 15 mm (0.6 in.) diameter or 30 mm (1.2 in.) diameter and the path of each cutter will vary as indicated. Similarly, the profile of the component shown in Figure 3.25 could be turned using a tool with a tip radius of 1 or 2 mm (0.04 or 0.08 in.) and again the cutter paths will vary.

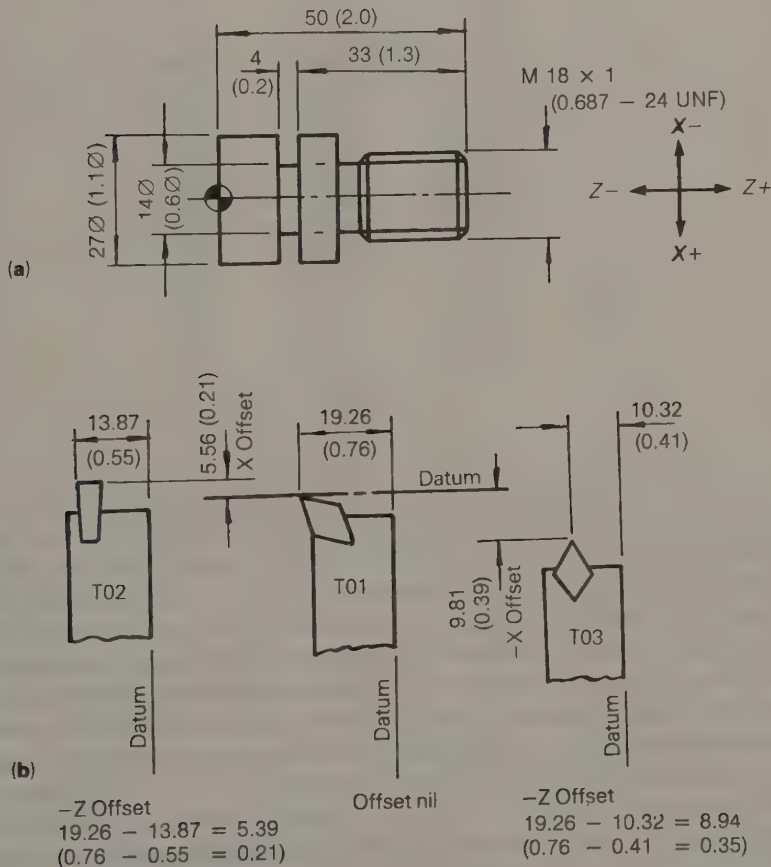


Figure 3.22 (a) Component detail and (b) tool length offsets for a turning operation. (Inch units are given in parentheses.)

	T	X	Z	Radius
	1	0.0000	0.0000	0.0800
	2	0.2100	-0.2100	0.0000
	3	0.3900	-0.3500	0.0600
	4	0.0000	0.0000	0.0000
	5	0.0000	0.0000	0.0000
(a)	6	0.0000	0.0000	0.0000
	7	0.0000	0.0000	0.0000
	8	0.0000	0.0000	0.0000
	9	0.0000	0.0000	0.0000
	10	0.0000	0.0000	0.0000
	11	0.0000	0.0000	0.0000
	12	0.0000	0.0000	0.0000
	13	0.0000	0.0000	0.0000
	14	0.0000	0.0000	0.0000

	T	X	Z	Radius
	1	0.0000	0.0000	2.0000
	2	5.5600	-5.3900	0.0000
	3	-9.8100	-8.9400	1.5000
	4	0.0000	0.0000	0.0000
	5	0.0000	0.0000	0.0000
	6	0.0000	0.0000	0.0000
(b)	7	0.0000	0.0000	0.0000
	8	0.0000	0.0000	0.0000
	9	0.0000	0.0000	0.0000
	10	0.0000	0.0000	0.0000
	11	0.0000	0.0000	0.0000
	12	0.0000	0.0000	0.0000
	13	0.0000	0.0000	0.0000
	14	0.0000	0.0000	0.0000

Figure 3.23 (a) Inch and (b) metric tool offset file for turning.

Without a cutter radius compensation facility the programmer would have to state the precise size of the cutting tools to be used and program the machine slide movements accordingly. With the facility the cutter size can be ignored and the work profile programmed. The exact size of the cutting tool to be used for machining is entered by the operator into the offset file and when the offset is called into the program, automatic compensation in slide movement will be made. Radius compensation also would allow the programmer to program nominal sized cutters and compensate for tool diameter variations or effects of machining cutter deflections. This type of programming may be necessary owing to limits on the amount of compensation available.

For milling machines the cutter size is entered as a diameter, in the example shown in Figure 3.21, and the machine slide movement is compensated by half of the dimensional entry. Note, depending on the control, the information could be entered as a radius with full amount of compensation occurring. For turning

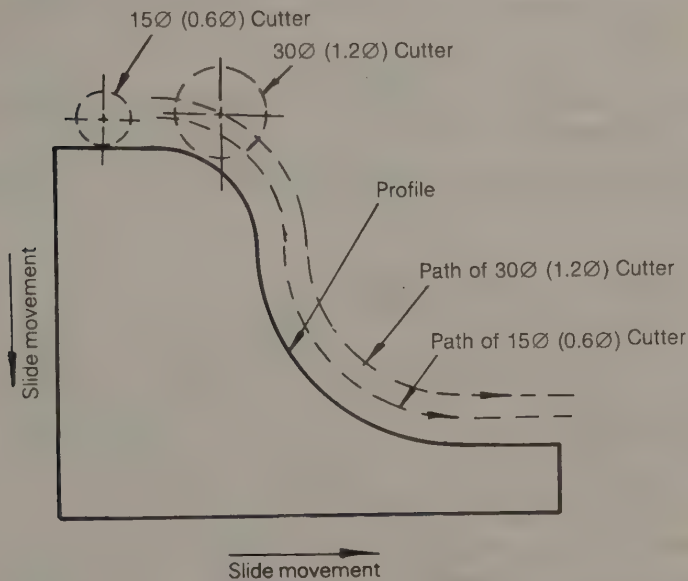


Figure 3.24 Cutter radius offset for milling operation. (Inch units are given in parentheses.)

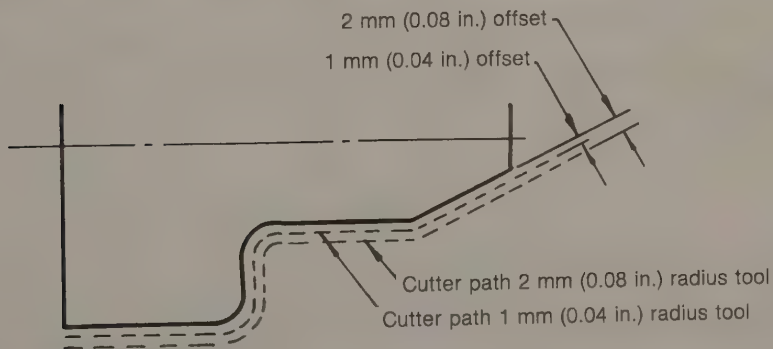


Figure 3.25 Cutter radius offset for turning operation. (Inch units are given in parentheses.)

centers the offset will be entered as a radius alongside the tool length offsets, as shown in Figure 3.23.

Cutter radius compensation can be to the right or left of a profile. To determine which is applicable the programmer should imagine a position above the tool facing the direction in which cutting is taking place. Thus cutter radius compensation to machine the profile shown in Figure 3.24 would be to the left. Compensation direction will be determined by a special program code discussed in the programming area.

Tool offsets can be entered, modified or erased by the machine operator at will and so it is possible to use the facility to:

- (a) accommodate replacement tooling which varies dimensionally from the original;
- (b) make variations to the component size;
- (c) initiate a series of cuts, say roughing and finishing, using the same dimensional program data.

While offsets have a direct effect on the machining currently being carried out, they do not affect the basic part program.

Identification of Cutter Offsets

Reference again to Figures 3.21 and 3.23 will show that the offsets are numbered. In a similar way tooling used in any part program is given a numerical identity. The two, tools and offsets, have to be related to each other when the part program is being made. The number of tool stations on any one machine is limited, perhaps 12 to 16 on a turning center to rather more on machining centers equipped with magazines. The number of offsets available will be greater than the number of tools available so that any tool can be used with any offset. Thus if the tools are numbered T01, T02, T03, T04, and so on, and the offsets are numbered 01 to 32, the programmer may call for tool number one to be used with offset number one. The data entry in the part program could read T0101, sometimes special codes like H01 may call the offset active. It follows that, since there are more offsets available than tools, the program could well call for the same tool to be used elsewhere with yet another offset, say T0106. It is imperative that the programmer's intentions are clearly relayed to the shop floor. (See Chapter 8.)

TOOL CONTROL

The efficient use of expensive computer numerically controlled machining facilities requires a very methodical approach to the provision of tooling. It is essential that the tooling, both original and replacement, available at the machine correspond to the tooling required by the part program. Close cooperation between personnel concerned with programming, tool preparation, and machining must be maintained.

Efficient tool control should provide for the following functions:

- (a) reconditioning, including regrinding when appropriate, replacing damaged or worn inserts, etc.;
- (b) preparation, including sizing, presetting, identifying, etc.;
- (c) storage until required for use;
- (d) transportation;
- (e) storage alongside the machine.

The concept is illustrated diagrammatically in Figure 3.26.

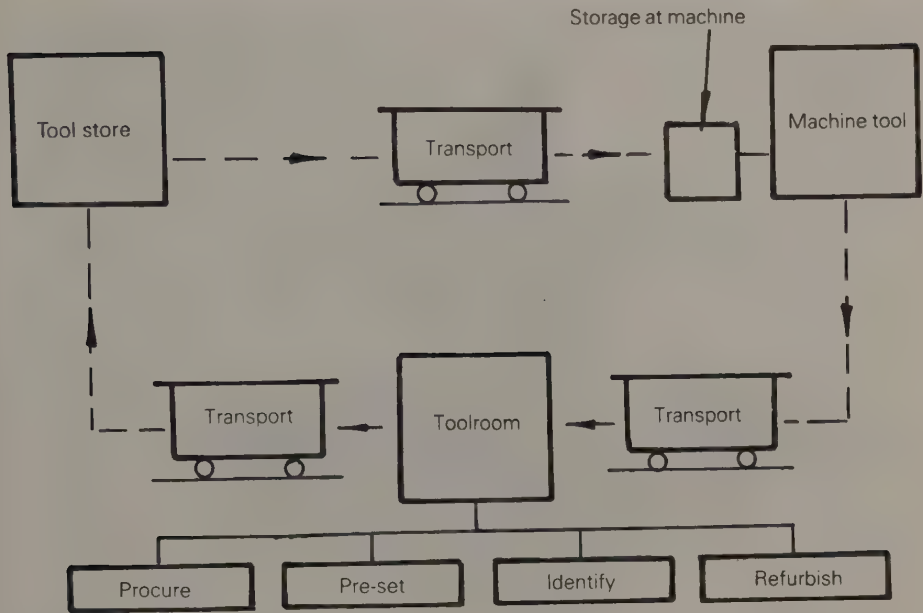


Figure 3.26 *Tool control system.*

Tools that can be reconditioned by grinding require skilled attention. Cutting efficiency over long periods of machining at very high rates of metal removal demands exact tool geometry. The less than precise methods often applied to tooling used on conventional machines are not acceptable for computer numerically controlled machining.

The accuracy demanded and the nature of the equipment used when presetting or establishing the precise size of tools call for skilled personnel working in clean conditions.

As tooling is prepared it must immediately be marked to facilitate identification, which can be done in a number of ways. Identification will of course correspond to the tool identity allocated by the part programmer and the tool preparation will be based on instructions he or she has prepared.

The storage of tooling not immediately needed requires the use of heavy-duty steel racks in which the tools are identified as to future machine magazine location.

Stored tooling may be for a specific job or it may be for general use. Either way, the tooling available should be fully documented regarding its dimensional features, application, etc. It is also helpful if the available tooling is listed in a manner that provides a ready reference facility for part programmers, machine setters, and others associated with the practicalities of the production process. The tooling list itself is often referred to as the "tool library."

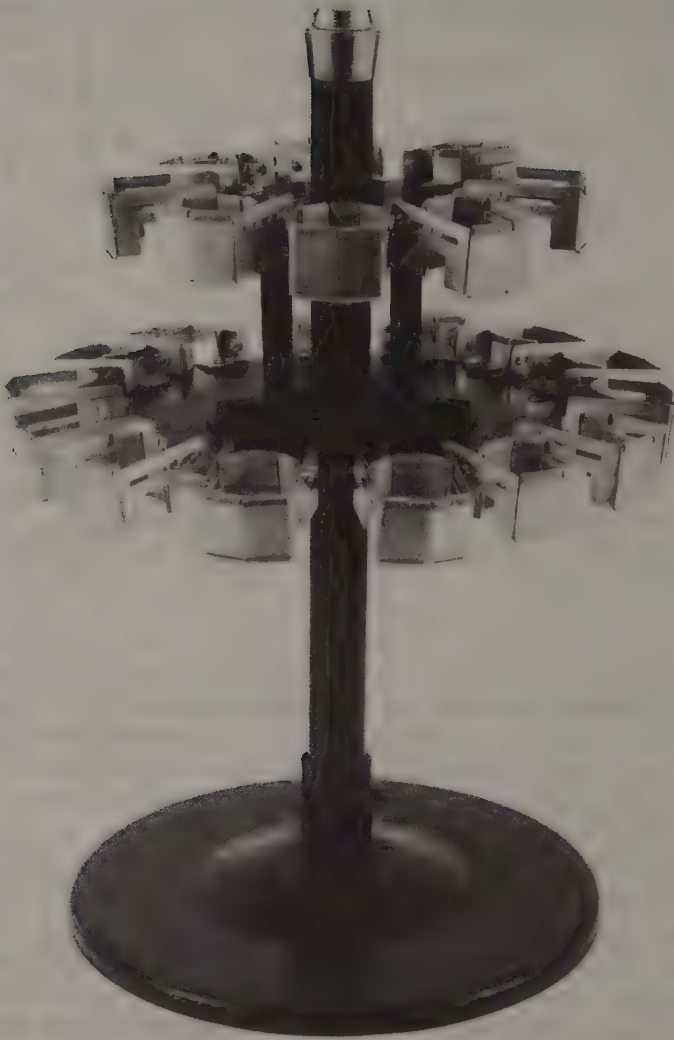


Figure 3.27 *Tool storage wheel.*

Transport of sets of tooling about the plant will require suitable carts. In some cases, to reduce handling, such carts are used for storage at the machine. If space on the shop floor is restricted, transfer to a stand may be more appropriate. One example of such a stand is referred to as a “tool wheel,” since it rotates to facilitate access to each tool; this is shown in Figure 3.27.

QUESTIONS

- 1 When are high-speed steel-cutting tools likely to be used in CNC machining?
- 2 When are solid carbide tools, as opposed to tips, likely to be used in CNC machining?
- 3 Why are solid carbide tools not widely used?
- 4 State three advantages of using indexable inserts.
- 5 State two methods used to control chips when using indexable inserts.
- 6 What is the significance of the letters P, M, and K in relation to the classification of carbide grades?
- 7 How are carbide inserts classified?
- 8 What is the difference between a holder and a cartridge?
- 9 Make an outline sketch of a tool suitable for use on a turning center and explain the meaning of the term "qualified."
- 10 How many methods of locating and clamping inserts in holders and cartridges are included in the ISO code?
- 11 The following is the specification for a tool holder: M P D F L 40 40 D 24 F. What is the meaning of each letter and number?
- 12 Explain the difference between a tool turret and a tool magazine.
- 13 If a machining program is to remain valid when tool replacement is carried out, one of two conditions must be met. What are those conditions?
- 14 Explain with the aid of a simple diagram what is meant by preset tooling.
- 15 What is a tool length offset and when is it likely to be necessary?
- 16 What is cutter radius compensation, and how does it simplify programming?
- 17 How is it possible to determine whether cutter compensation is to the right or left of a machined profile?
- 18 List the functions of an efficient tool control system.
- 19 What is the function of a tool library?
- 20 What is a tool wheel and when is it used?

4

WORK HOLDING AND LOADING FOR COMPUTER NUMERICALLY CONTROLLED MACHINING

THE APPLICATION OF COMMON WORK-HOLDING DEVICES

The basic requirements of any work-holding device are that it must

- (a) securely hold the work;
- (b) provide positive location;
- (c) be quick and easy to operate.

There are a variety of devices in general use that have been tried and tested in conventional machining situations. Chucks, collets, and vices are obvious examples, and these are also used on computer numerically controlled machines. Work-holding devices such as these may be mechanical, pneumatic, or hydraulic in operation. Mechanically operated devices usually involve manual intervention and, although it is not uncommon to see workpieces being loaded and clamped in this way, it is not a practice that is in keeping with automatic machining processes. Because of this, hydraulically or pneumatically operated devices, especially the latter, are favored. The operation of hydraulic or pneumatic clamping is easily controlled electronically via the machine control unit and also provides for rapid operation and uniform clamping pressure. The application of a power-operated collet is shown in Figure 4.1 and a power-operated chuck is shown in Figure 4.2.

Conventional devices such as these are more suited to machining where the component or the stock material is uniform in shape, that is, rectangular, round, hexagonal, etc. Components of irregular shape, such as castings, can be accommodated, as with conventional machining, on specially built fixtures sometimes incorporating pneumatic or hydraulic clamping arrangements.

THE IMPORTANCE OF ACCURATE LOCATION

It is established working practice that, wherever possible, work should be positively located, that is, it should be positioned in such a way that, when the cutting forces are applied, no movement can take place.

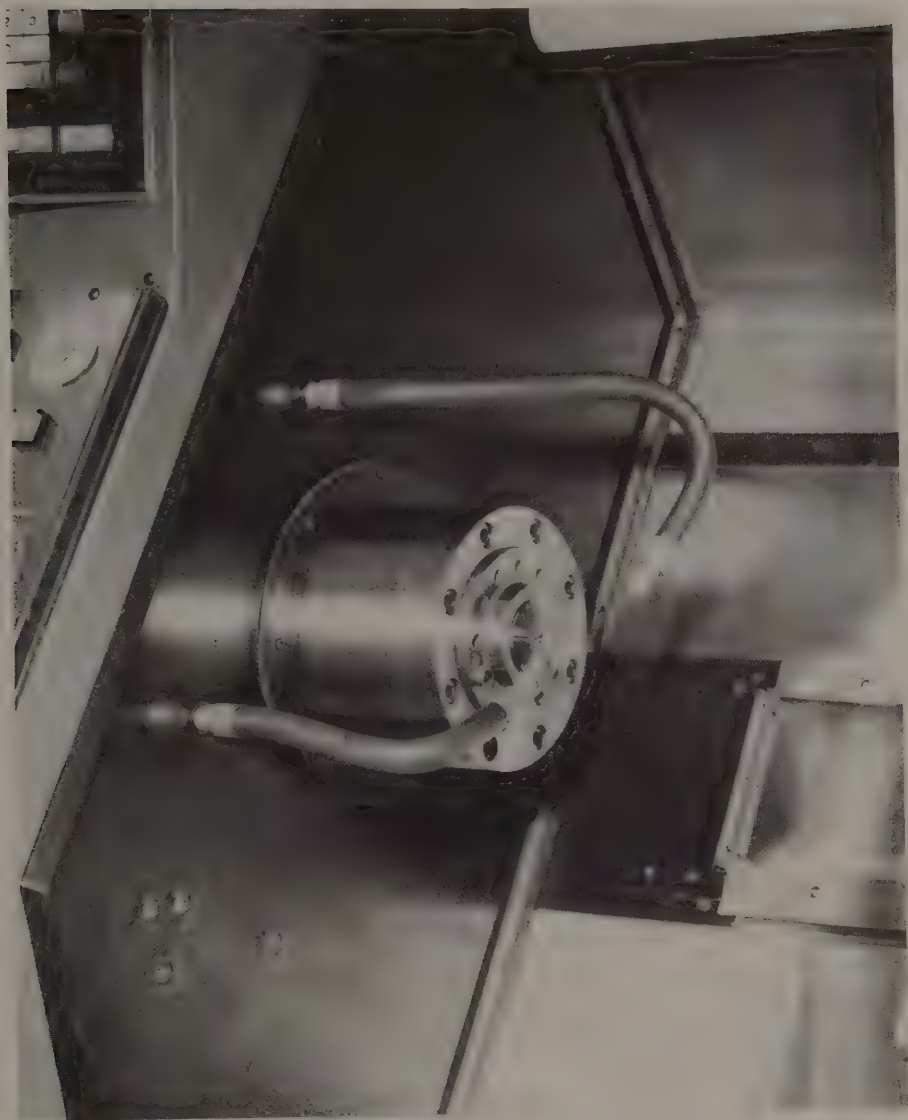


Figure 4.1 *Use of collet for work holding on a turning center.*

Figure 4.3 shows two applications of a conventional machine vice. In both cases the work is located against the fixed jaw but in Figure 4.3a the security of the workpiece depends on a frictional hold and the cutting force could result in movement of the workpiece. In Figure 4.3b no movement is possible since the fixed jaw of the vice not only locates the workpiece but also absorbs the forces resulting from the cutting action.



Figure 4.2 *Use of power-operated chuck on a turning center.*

Similarly, in Figure 4.4a it can be seen that it is possible for the workpiece held in the chuck to move, since it is not positively located. Figure 4.4b shows how the possibility of movement is eliminated by using the back face of the chuck for positive location.

In any machining process the possibility of movement of the workpiece is unacceptable for safety reasons. In computer numerically controlled machining processes there is also the problem that movement, however slight, means a loss of dimensional accuracy, since there is generally no constant monitoring of the workpiece size as machining proceeds. Additionally, the location of the component is often directly related to the part program, since the programmer, when writing the program, will establish datums on which all numerical data controlling the machine slide movements will be based. If the component is not precisely positioned in relation to those datums, then the machining features required will not be achieved.

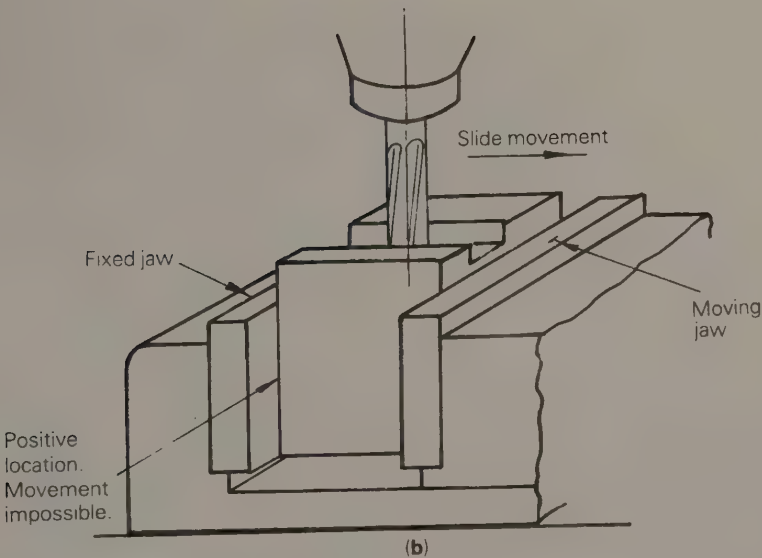
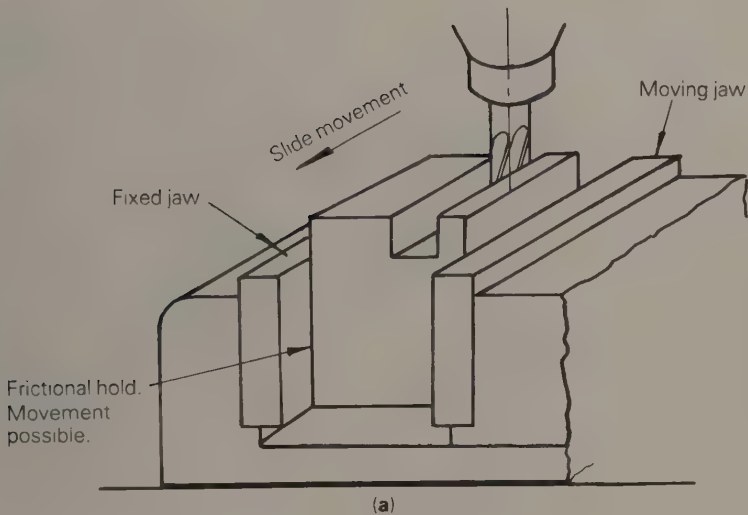


Figure 4.3 Positive location of a milled component: (a) unsatisfactory and (b) satisfactory.

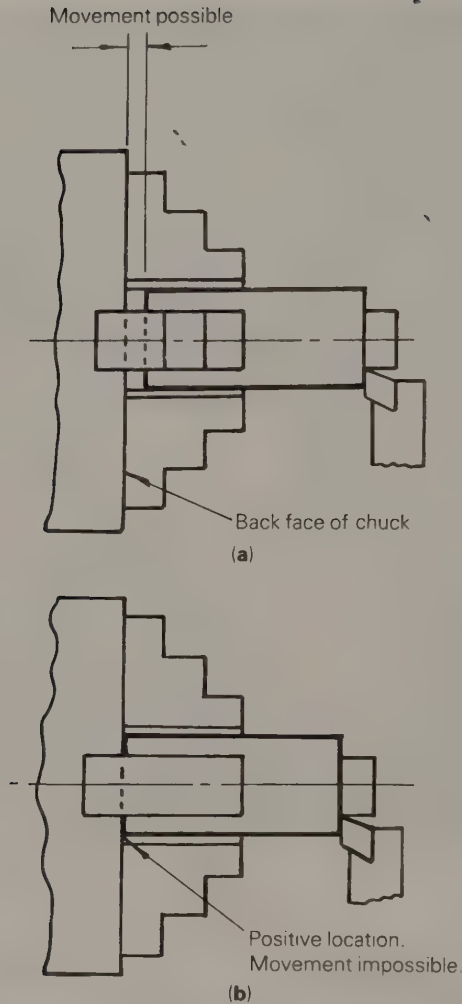


Figure 4.4 Positive location of a turned component: (a) unsatisfactory and (b) satisfactory.

Figure 4.5a shows a component that is to be machined on a vertical machining center and the datum in the X and Y axes that the programmer has established as a basis for the part program.

The machine setter will be informed of the position of the datum, either by written instructions or possibly by messages included as part of the program and visually displayed on the machine control unit, and he or she will be required to set the work-holding device, which in turn provides the precise location for the component, accordingly.

To illustrate the need for accurate component location, again consider Figure 4.5. Clearly a workpiece positioned as shown in (b) will not have the same

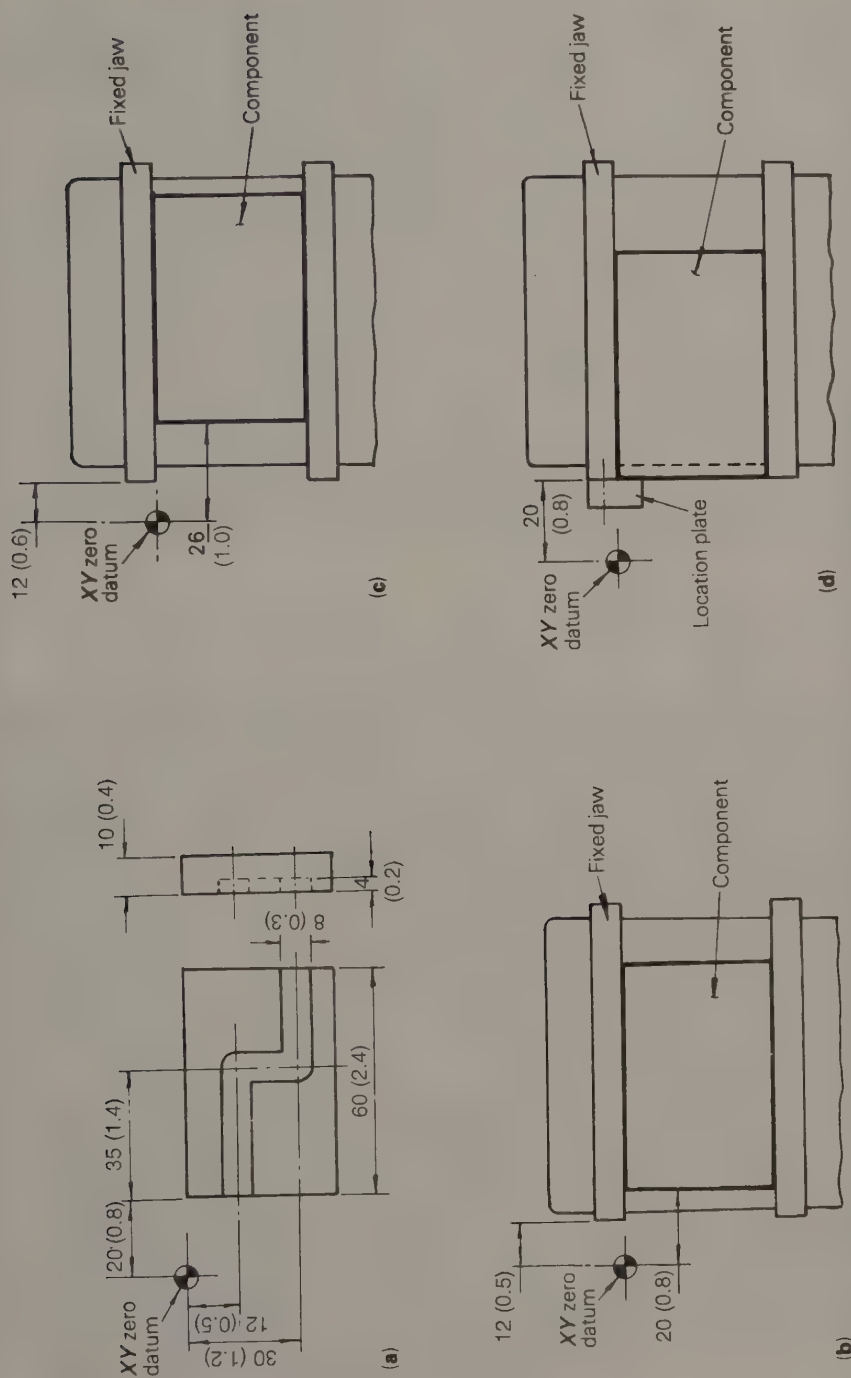


Figure 4.5 Relationship between setting position and part program for a milled component: (a) component detail; (b) initial setting; (c) second setting; (d) setting position constant, no variation in component size. (Inch units are given in parentheses.)

dimensional features in the X axis as the component position shown in (c). In this particular case the logical thing to do would be to place all components as shown in (d), thus using the end of the vice jaw as a locating position, and set the machine datum accordingly. A stop plate attached to the side of the fixed jaw is a method which can be used to ensure perfect location.

Figure 4.6 shows the use of a self-centering chuck where the back face provides the datum and location face in the Z axis and the self-centering action provides the datum and location in the X axis. The positioning of the workpiece to establish positive location in the X axis is automatic, but the machine setter will need to be informed that the work is to be located against the back face of the chuck to maintain the dimensional validity of the part program in the Z axis.

When the bar size is smaller than the machine spindle bore, accurate location may still be achieved by using a special part stop placed inside the chuck or spindle and against which the component is located prior to clamping. The internal stop is not removed before machining commences so the positive location is maintained. An alternative to using an internal stop is to use soft jaws bored to suit, with a shoulder acting as a stop.

THE USE OF GRID PLATES FOR MILLING AND DRILLING

A method of work holding and location that has gained wide acceptance for computer numerically controlled milling and drilling setups is the grid plate.

A grid plate is simply a base plate made of steel or cast iron that is drilled with a series of accurately positioned holes. These holes may be tapped to facilitate clamping, plain reamed to accommodate location dowels, or tapped and counterbored to provide for clamping and location. Each hole can be identified using the grid system illustrated in Figure 4.7.

The grid plate is attached to the machine table, often permanently, and since the part programmer can identify the exact position of any hole and will know the dimensions of any locating dowels or blocks used in the work-holding arrangement, he or she can establish datums when writing the program and instruct the machine setter accordingly.

The setting of a grid plate does not involve the use of dial indicators, edge finders, etc., and therefore is not demanding on manual skills on the shop floor. Once set, it provides for quick, simple, and accurate location of the workpiece.

It is often possible to load more than one component at each setting and at known pitches. By using the "zero shift" facility (see Chapter 6), the machining program can be repeated in a new position with a resulting saving in machine downtime.

Apart from clamping directly to the grid plate, components can be held in fixtures, vises, or a set of vises and fixtures, which themselves are accurately located and clamped in position. Complex shapes can be accommodated by

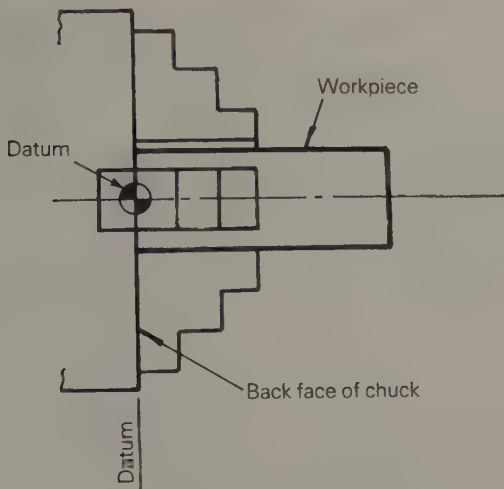


Figure 4.6 Accurate location of a turned component providing constant datum position.

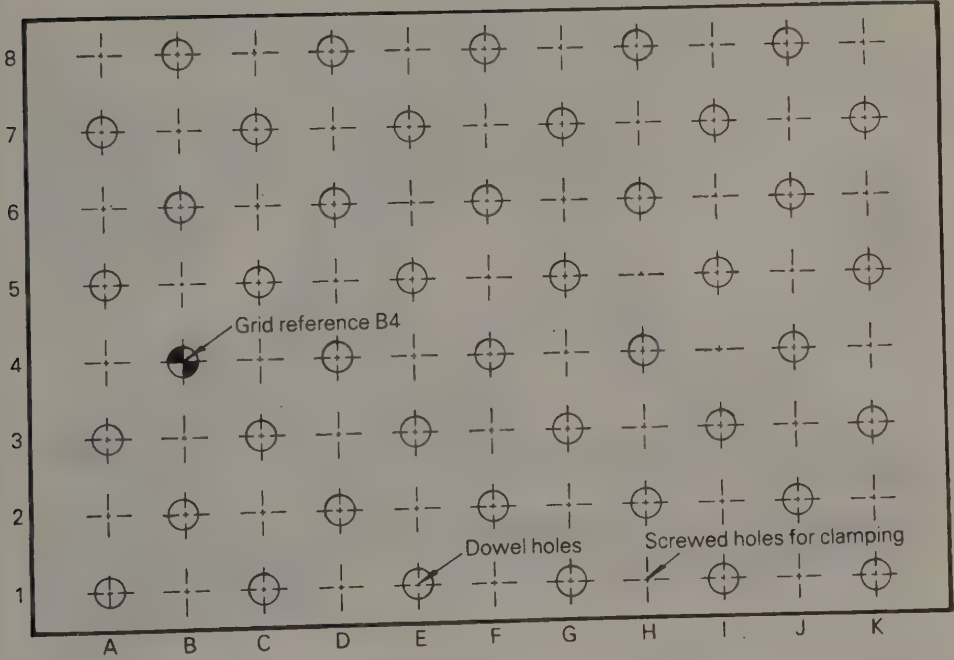


Figure 4.7 Principle of grid plate location.

using specially shaped locators, while fixtures can be provided with locating devices so that they may be accurately located and clamped in a known position.

Through cutting is possible by using stepped locators that raise the workpiece from the grid plate.

Examples of commercially available grid plates for both horizontal and vertical applications are shown in Figures 4.8 and 4.9.

THE USE OF ROTARY TABLES AND INDEXERS FOR MILLING AND DRILLING

Many of the conventional uses of a rotary table have become redundant with the introduction of numerical control. Radial profiles are now achieved by circular interpolation, and the positioning of holes or slots in angular relationship to each other, possibly using polar coordinates, has been reduced to nothing more complex than a simple one-block data entry in the machine program. Circular interpolation and polar coordinates are discussed in more detail in Chapter 6.

Rotary tables are still used on horizontal machines for rotating work to fa-

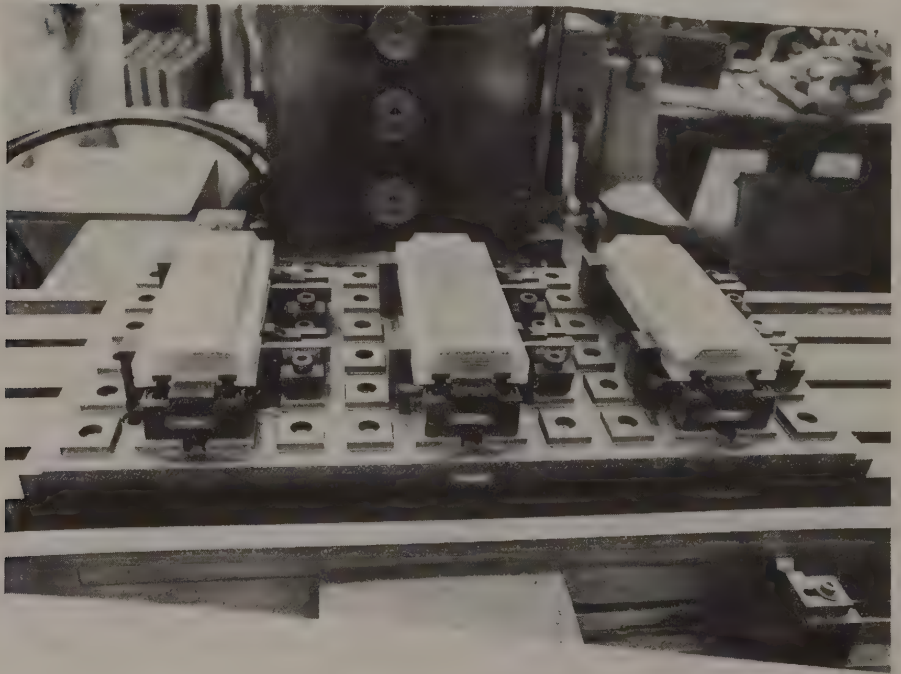


Figure 4.8 Grid plate showing three components in position.

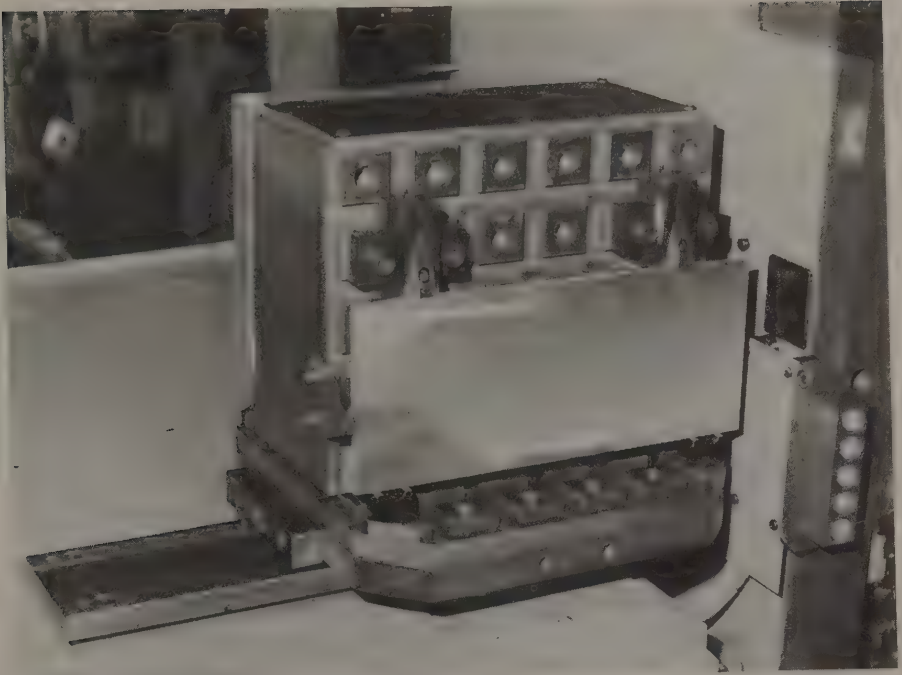


Figure 4.9 *Vertical application of a grid plate.*

cilitate machining in a new position in the vertical plane. An angle of rotation of 90° , for example, permits machining on four sides of a cube, but the indexer can be used to rotate the work through much smaller angles. An angle of rotation as small as 1° or 360 circular positions are common.

Rotary tables of this type may be attached to the machine bed in the normal way or be a built-in feature of the machine table, as illustrated in Figure 4.10.

Conventional dividing heads are also redundant as far as computer numerical control is concerned. They have been replaced by indexers, fully programmable and controlled via the machining program. Simple versions allow up to 24 positions, or increments of 15° , rather like the direct indexing plate fitted to conventional dividing heads.

For more complex indexing or where continual rotation is required, for example when cutting a helix, a rather more sophisticated version is needed, with up to 360,000 positions and feedrate controls. Some of these devices are capable of rotating in two planes. (Rather confusingly, they are referred to as "tilting rotary tables" by one manufacturer.) When the two-axis version is used in conjunction with three axes of table movement, thus providing a five-axis machining capability, it permits the production of components so complex that they may well be incapable of being produced by conventional means.

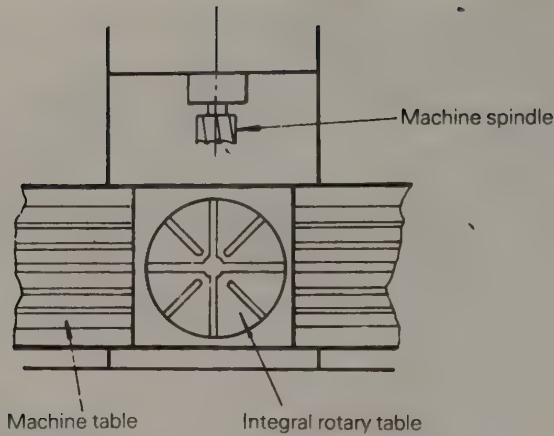


Figure 4.10 *Integral rotary table.*

THE USE OF PALLETS FOR MILLING AND DRILLING

An aim of production engineers is to minimize downtime, that is, the time when the machine is not fulfilling its prime function of cutting metal. A major source of downtime is work loading and unloading. The use of preloaded pallets considerably improves the situation.

A pallet is simply a table, which, like the grid plate, is provided with a series of holes or slots to facilitate location and clamping of the component. Pallets are fitted to the machines, shown in Figures 2.1b and 3.14, and can be shuttled in and out of the machining area.

The most simple arrangement will involve the use of just two pallets. A workpiece is located and clamped on the first pallet in a position predetermined by the part programmer and the pallet is then moved into the machining position. As machining is taking place, the second pallet is loaded. When machining of the first component is complete, the pallets are interchanged and, as the second component is being machined, the first pallet is unloaded and reloaded with another component.

Pallets can be interchanged in several ways. Two such methods, one involving a shuttle system and the other a rotary movement, are illustrated in Figures 4.11a and 4.11b. Some machining systems involve more than two pallets (see Chapter 9).

WORK SUPPORT FOR TURNING OPERATIONS

Most turned components are relatively small. The work-holding arrangements used in their production are conventional, that is, chucks and collets are used

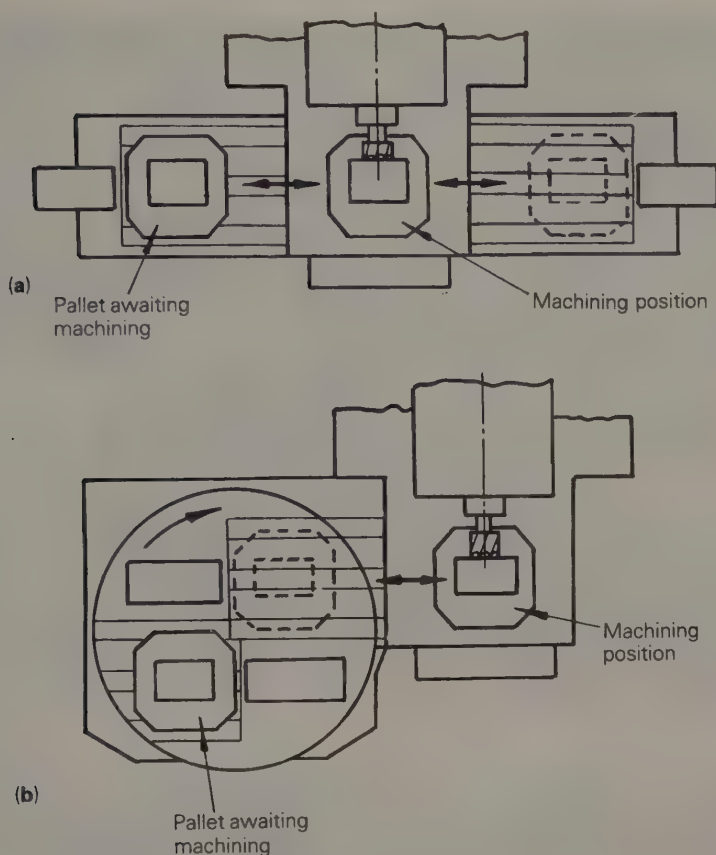


Figure 4.11 Pallet shuttles: (a) linear and (b) rotary.

and there is no need for further work support. Because of this, a number of the smaller turning centers available do not have tailstocks. They are no longer essential for drilling, reaming, etc., as this work is carried out from the turret.

When the capacity of the machine is such that the work overhang can be considerable, then tailstock support becomes essential. It is also necessary, of course, for turning between centers. Figure 4.12 illustrates a tailstock being used on a computer numerically controlled turning center.

On some machines the tailstock is very similar to that of a conventional center lathe. It is positioned and clamped solely by manual intervention. On others, it is partially controlled, being manually positioned and clamped on the machine bed but with a programmable hydraulic quill movement. Some machines provide for a fully programmable tailstock, that is, both its position along the bed and the quill movement, together with the necessary clamping, can be included in the part program and automatically controlled.

If a fully programmed tailstock movement is to be used, it is essential that,

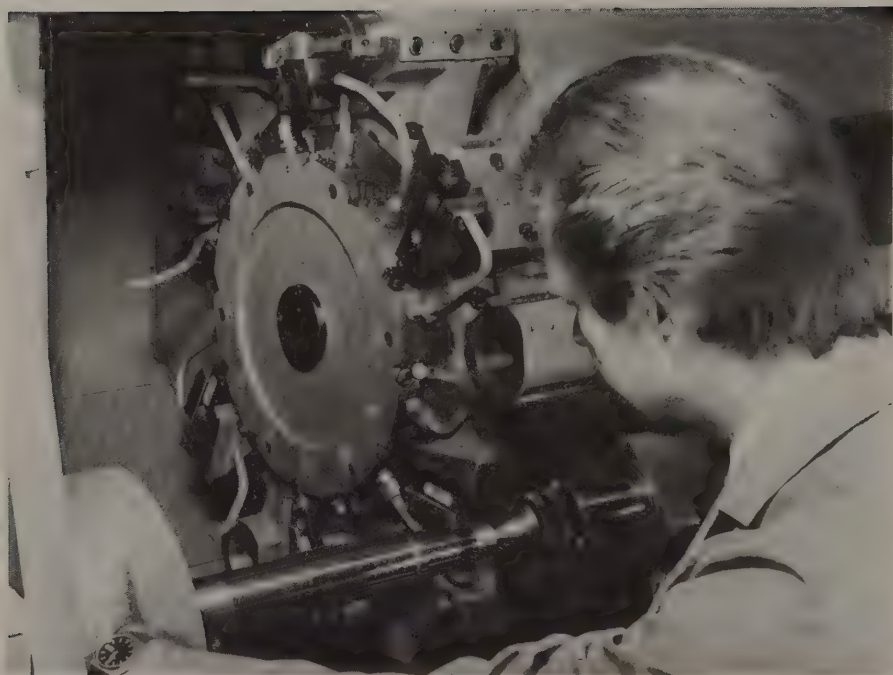


Figure 4.12 *Use of a tailstock for work support on a turning center.*

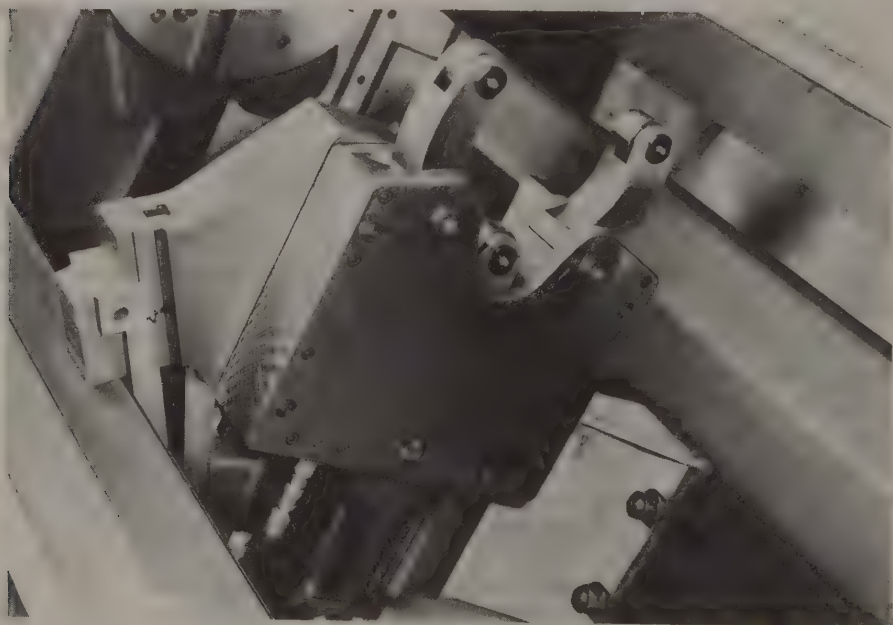


Figure 4.13 *Bar steady rest providing pressure-controlled support.*

as the quill moves forward, the workpiece is in the correct position, that is, on center in order to receive the center support. Acknowledging that this condition may be difficult to obtain, some manufacturers offer hydraulic self-centering steady rest to position the work prior to the tailstock movement being made.

Steady rests are also available to prevent the deflection of slender work, these being located on the machine bed in a manner similar to the way steady rests are used on conventional lathes. Two types of steady rests are illustrated in Figures 4.13 and 4.14.

WORK LOADING FOR TURNING OPERATIONS

Work loading into turning centers may be manual or automatic. The choice of method to be used will be affected by various factors such as component size, component shape, and quantity required.

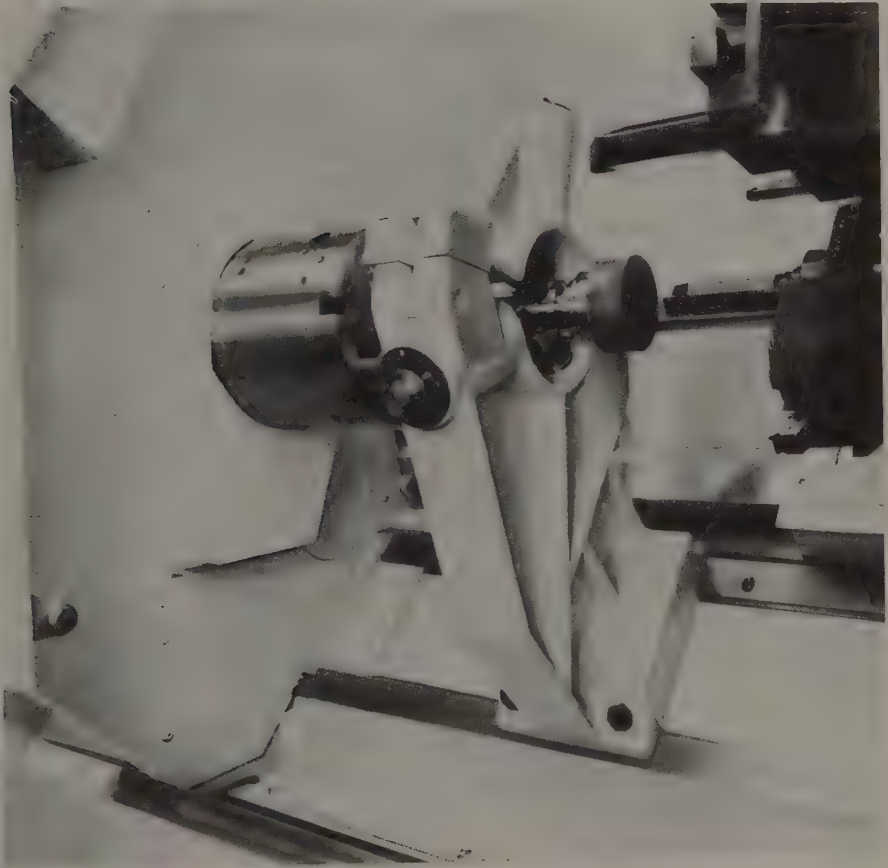


Figure 4.14 *Mechanically adjusted bar steady rest.*

Manual loading detracts from the benefits in terms of increased production rates and reduced labor costs inherent in computer numerically controlled machining. However, it is quite acceptable for small-batch production and indeed may be essential when the component shape is irregular, for example, a casting, or when nonstandard work-holding devices, such as a fixture clamped to a face plate, are being used.

When automatic loading is applicable, the cost of the necessary equipment is likely to be the determining factor in the final choice. The possibilities range from relatively inexpensive bar feeders to arrangements involving conveyors and robots.

Bar feeders have been applied to turning machines for many years. One of the disadvantages associated with earlier designs was that they were noisy in

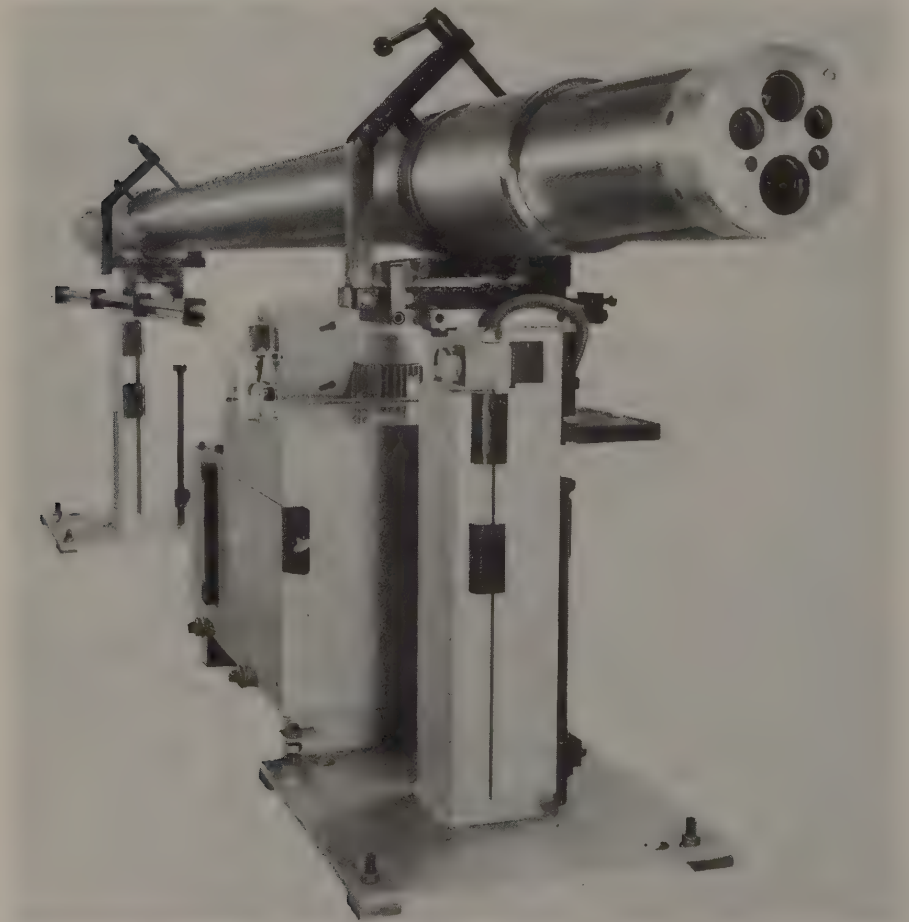


Figure 4.15 *Silent bar feeder. Bar is supported in oil.*

operation, the noise being created by the bar rotating in the feeder tube. Modern designs have eliminated this problem by various methods, perhaps the most successful one being where the bar is completely surrounded by oil. The bar is fed into the machine under pressure (hydraulic and pneumatic systems are available) when the work-holding device releases its grip. The bar extends to a preset stop located in the turret before the work-holding device closes again.

Bar feeders do not provide total automation, since they have to be reloaded manually from time to time. A modern bar feeder is shown in Figure 4.15.

Components that are too large for bar feeders to be suitable are often produced from preprepared "billets," that is, the material is supplied in short lengths, sometimes already faced to size. Material in this form, and partly machined components of similar size requiring further machining, are usually suitable for robot handling. Many machine manufacturers offer robot handling equipment as an optional extra, the robot being adaptable to various component shapes and sizes by fitting interchangeable end effectors (grippers).

QUESTIONS

- 1 Explain what is meant by positive location as applied to work holding.
- 2 Why is it especially important that components are positively located for computer numerically controlled machining operations?
- 3 Explain how a workpiece can be positively located in a self-centering power-operated chuck.
- 4 State three reasons why pneumatic or hydraulically operated work-holding devices are particularly suitable for use on computer numerically controlled machines.
- 5 List the advantages of the grid plate as a means of holding workpieces.
- 6 Describe how the positioning of a workpiece on a grid plate is identified.
- 7 How would components of irregular shape, such as castings or forgings, be held on a grid plate?
- 8 Much of the work carried out on a rotary table using conventional machines is achieved in other ways on computer numerically controlled machines. Quote examples where the facilities provided by a rotary table are still useful.
- 9 What is the main advantage of using preloaded pallets?
- 10 Describe two ways in which pallets are interchanged on a machining center.
- 11 Why is the accurate location of a pallet essential before machining commences?

- 12 Explain what is meant by a fully programmable tailstock as used on a turning center and briefly describe the alternative types of tailstock.
- 13 What is the main disadvantage of having manual work loading in a computer numerically controlled machining situation? Give an example of a situation where there is unlikely to be an economically viable alternative.
- 14 What are the advantages and disadvantages of using bar feeders for turning centers?
- 15 What are billets and what are the advantages in their use?

5

DATA PREPARATION AND INPUT TO MACHINE CONTROL UNITS

DATA PREPARATION

The preparation of numerical data prior to input to the machine control unit is referred to as programming. The extent of the preparation will depend on the complexity of the component. It is possible that the data necessary to produce a simple component may require nothing more than an examination of a detailed drawing followed by a direct manual entry to the control unit. On the other hand, programming very complex components may require computing facilities to determine appropriate tool paths. The vast majority of components require an approach similar to that outlined diagrammatically in Figure 5.1.

From the diagram it can be seen that the program is central to the whole process. It is compiled after taking into account a number of essential inter-related factors and then, having been compiled, it totally controls the machining process. Efficient programming requires considerable practical knowledge on the part of the programmer together with a full understanding of the control system to be employed.

The approach to programming must be methodical, and because of this it usually involves compiling a special form or listing the data on a computer screen, preferably followed by a checking process, before recording the data in a form acceptable to the machine control unit. Even at this late stage a further checking process, referred to as "program proving," is essential before a final commitment to machining is made.

DATA INPUT

Data can be entered into machine control units by the following methods:

1. Manual data input (MDI).
2. Conversational manual data input.
3. Perforated tape.
4. Magnetic tape.
5. Portable electronic data storage unit via an interfaced computer.
6. Magnetic disk via an interfaced computer.
7. Master computer (direct numerical control, DNC).

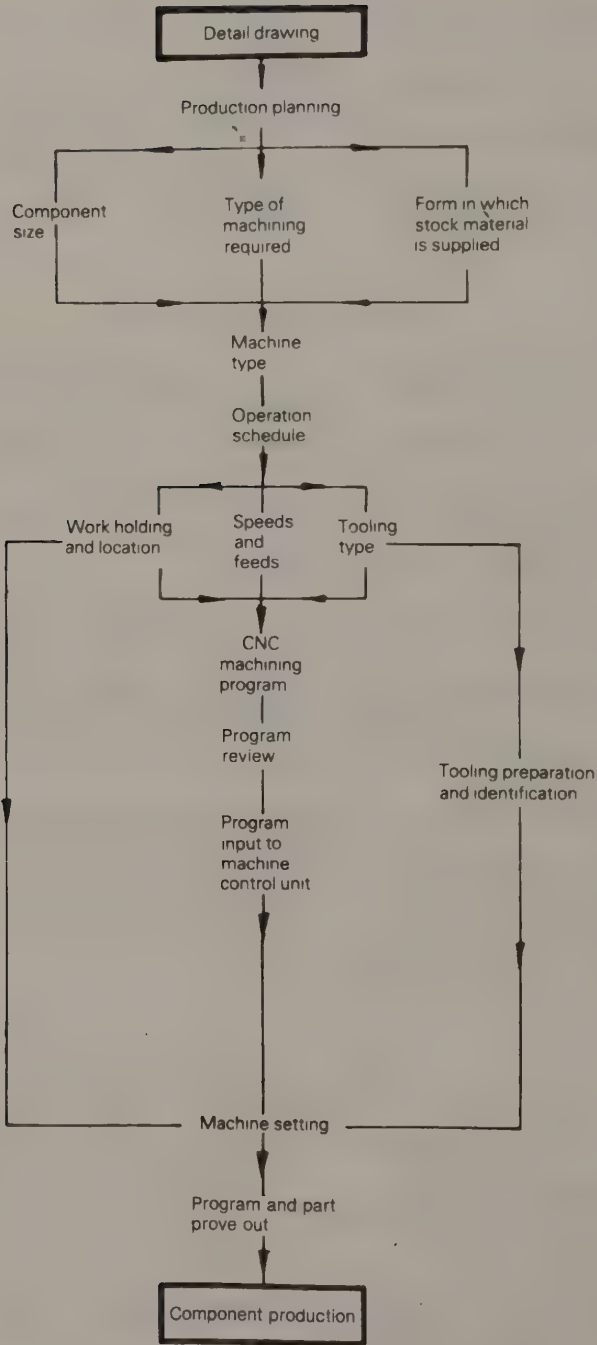


Figure 5.1 Approach to data preparation.

Manual Data Input

Manual data input is used when setting up the machine and editing programs, and for entering complete programs, although the latter should be restricted to relatively simple programs so that the machine is not idle for too long as the data entry is being made.

Manual Data Input to NC Machines To input data on noncomputerized control units the operator has to set dials, position switches, etc., before finally activating the machine tool to carry out the required movements. Only a limited amount of data can be entered at any one time. Data-recording facilities are often not available.

Manual Data Input to CNC Machines On computerized control units, by pressing the appropriate buttons on the control console a limited amount of data or a complete part program may be entered and the machine activated accordingly. The computer will retain the data and it can be transferred to a recording medium such as magnetic tape or disk and transferred back to the computer as and when required.

Conversational Manual Data Input

Conversational manual data input involves the operator pressing the appropriate keys on the control console in response to questions in everyday English, which appear on the visual display unit (CRT, cathode ray tube) screen. This method of manual data input is quicker than methods requiring the use of data codes, and manufacturers of these control units claim that to make the first chip takes one tenth of the time and that operator training is just a matter of hours as opposed to up to two weeks for nonconversational input.

The basis of conversational data input is the preprogramming of the computer with standard data stored in files within the computer memory, each item of data being numerically identified and called into the program by the appropriate operator response. Some machines maintain files of data from which the operator makes his or her final selections; other machines only ask questions to remind the operator what is needed.

Consider the turning of a bar of metal on a turning center. Before any consideration can be given to slide movements, the basic metal-cutting data would have to be ascertained. For example, the correct spindle speed and feed rate are of vital importance. The spindle speed is affected by the work diameter and the cutting speed. The cutting speed is related to the material being machined. The feed rate would depend on the depth of cut, tool type, and surface finish required. From this it can be seen that the necessary data to machine the metal successfully can be related to four factors:

- (a) the material being cut;
- (b) the material diameter;
- (c) the surface finish required;
- (d) the tool type.

The computer will be programmed to select the appropriate spindle speed and feed rate from an input of information relating to these factors. To assist the input of information there will be a material file and surface roughness file within the computer memory, as shown in Figure 5.2.

Cutting tools available will also be numerically identified. A simple question-and-answer routine will extract from the computer memory all the necessary data to give the correct cutting conditions; the programmer then enters the axes calculations or parameters to finish the program. An example of a question-and-answer routine is as follows:

CRT question	Operator's keyed response
Material?	5 input
Material diameter?	2.0 input
Surface code?	4 input
Tool number?	8 input

The preceding illustrates just a small part of the total data input necessary to make a component, and, having established basic cutting conditions, the programmer would proceed to feed in additional data relating to slide movement, etc. However, slide movement data input can be reduced to a question-and-answer routine, even when the movements are complex, such as when machining a radius or cutting a screw thread.

Consider the production of a thread on a work diameter that has just been produced by the preceding data entry. The required data input may be restricted to the following questions:

- Thread root diameter?
- Lead?
- Number of starts?
- Start location?
- Finish location?

From this the control will determine the number of passes necessary, the depth of cut taken by each pass, and the feed rate needed to produce the required pitch. Even the spindle speed may vary automatically to allow for roughing and finishing cuts.

MATERIAL STOCK FILE		SURFACE ROUGHNESS FILE	
CODE	MATERIAL	CODE	Microinches
1	MILD STEEL	1	100
2	MED. CARBON STEEL	2	50
3	STAINLESS STEEL	3	25
4	CAST IRON	4	12.5
5	ALUMINUM	5	6.3

Figure 5.2 Material file and surface roughness file.

Routines such as the one described are commonly known as “canned cycles” and are not restricted to conversational MDI but may also form part of other programming systems. The use of such routines is described in more detail in Chapter 6.

Data entered in response to questions can be recorded, usually on magnetic tape, for future use. Some advanced conversational MDI systems incorporate the use of computer graphics (see “Graphical Numerical Control,” Chapter 8).

Perforated Tape Input

Not so long ago numerical control was generally referred to as tape control, an indication of the important part this input medium has played in the development of the technology. The expression is not quite so common as it was, but perforated tape is still widely used.

The basis of tape control is the transfer of coded information contained on a perforated tape to the machine control unit via a tape reader.

The standard tape width is 1 in./25 mm. Originally only paper tape was used, and it is still the most popular material, a factor very much in its favor being its low cost. It is available in rolls or precisely folded in a concertina or fanlike arrangement (Figure 5.3). The rolls are most commonly used, but the folded paper is possibly easier to store.

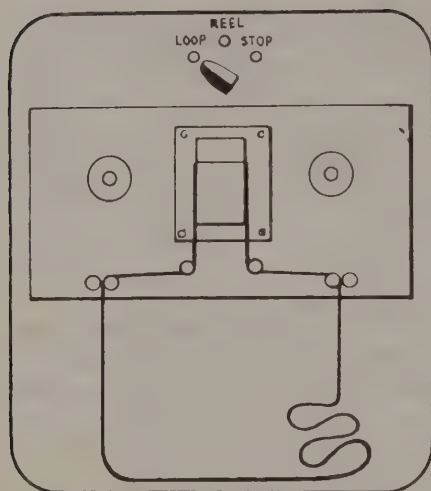
One of the problems with paper tape is that the sprocket drive holes that are used to carry the tape through some tape readers tend to wear or even tear. Also, a tape can easily be damaged by contact with oil, which is always likely in a shop atmosphere. This has led to the introduction of other materials or combinations of materials; but, while these tapes are more durable, they are usually more expensive.

Examples of tape materials other than paper include polyester film, paper-polyester-paper laminates, polyester-aluminum-foil-polyester laminates, metallized polyester, and aluminum laminate or Mylar. Some of these tapes can cause excessive punch wear and the adhesives used to produce some laminates have also presented punching problems.

The choice of tape is also affected by the tape reader being used. The prime function of the reader is to detect the presence and position of the perforations in the tape. Tape readers may be fiber optic or photoelectrical. Various tape readers are shown in Figure 5.4.

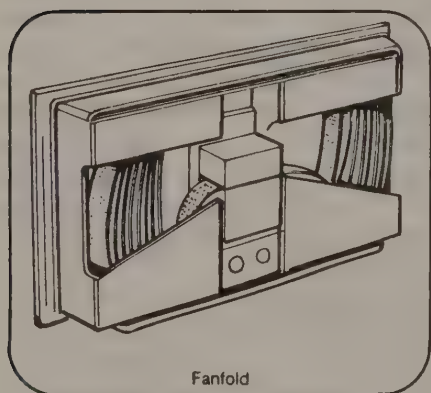
Fiber optic and photoelectric readers have light either passing through the perforations or being reflected off a reflector positioned behind the tape. Having detected a perforation, the reader converts this information into an electrical signal that is transmitted to the control unit. Light source readers using the reflective principle will require a tape with a nonreflective surface finish or color, while the direct light readers require tapes that are not translucent.

The qualities required of tapes used in photoelectric tape readers have resulted in a variety of colors being used. Some tapes are dual colored, which helps to reduce the possibility of reverse loading in the readers.



Loop

The tape is joined at its ends to produce a continuous loop.



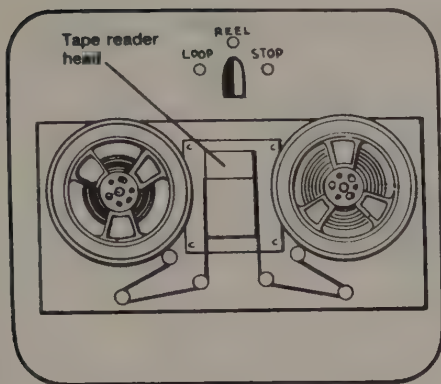
Fanfold or random length

The tape is drawn across the reader and passed from one compartment to the other.

Figure 5.3 CNC tape configurations.

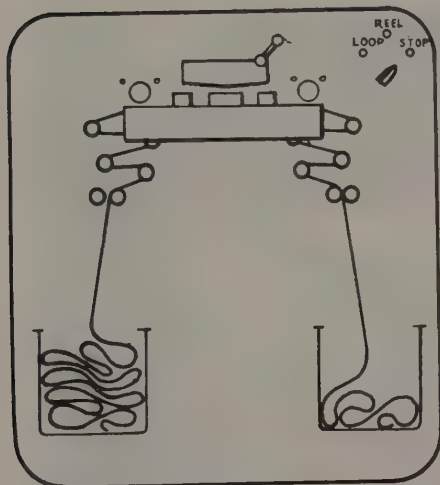
Reverse loading is also prevented by the offset of the sprocket holes when used, placing the edge of the tape closest to the sprocket holes to the back of the reader. In some cases the feed direction is indicated by arrows printed on the tape. In addition to this, when the tape is severed from the main roll after punching, the leading end will be pointed and the trailing end will have a corresponding recess.

The advent of the computer as an integral part of control systems has minimized the strength and wear problems associated with plain paper tapes. On older control units the tape is run through the reader each time a component is machined. For short programs it can be spliced to form a continuous loop, thus eliminating the need for rewinding. For longer programs the tape is wound



Reel

The tape is wound from one reel to another across the tape reader head.



Tape Tumble Box

The tape is left loose at both ends and is allowed to fall freely into a tumble box. Although this was an original configuration, it is rarely used today.

Figure 5.3 (Continued)

from reel to reel. Now it is common practice to use the tape more as a storage medium, feeding the data it contains into the control unit computer by just one pass through the tape reader. The computer retains the data in its memory and this facilitates data retrieval as and when required. When data are transferred from tape to the computer memory, the tape can be removed from the reader to avoid contamination.

Tape Standards and the Binary Code When a tape reader detects a perforation, the transmission of an electrical signal to the control unit results. The simplest way an electrical signal can be meaningful is by its presence or absence, creating an on-off effect. The detection of a perforation registers an "on" signal, and this signal will be given further meaning by the position of the perforation that caused it, as will be explained subsequently.

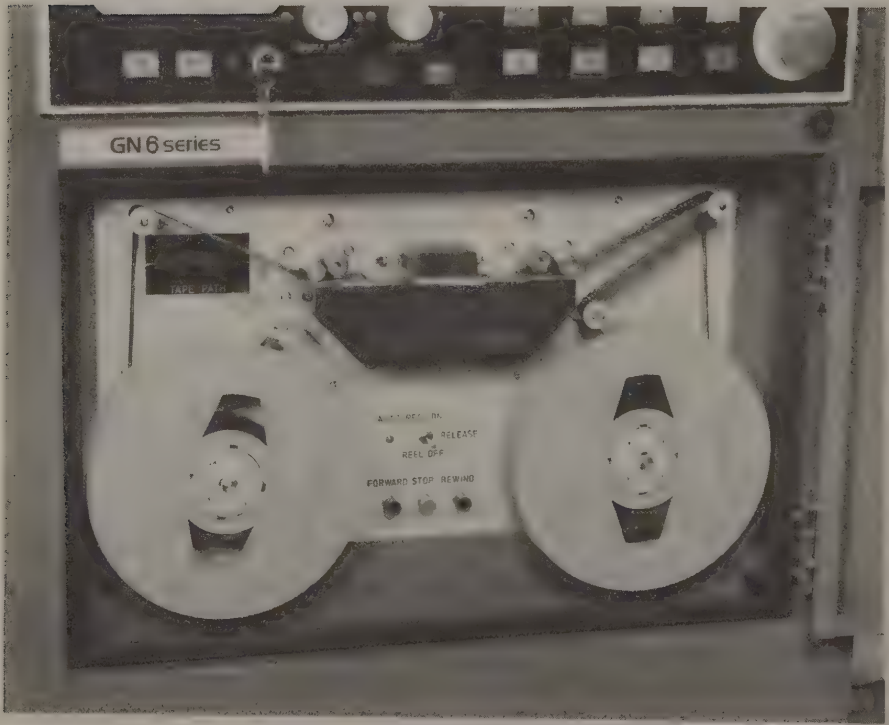
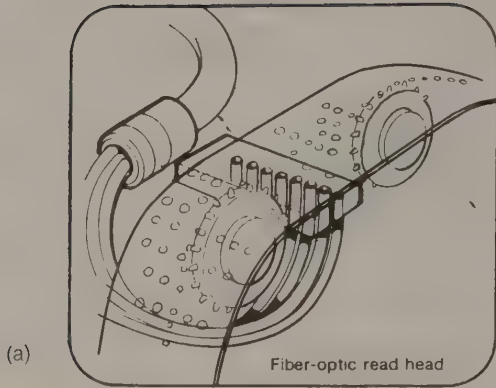


Figure 5.4 (a) Fiber-optic and (b) photoelectric tape reader.

Whatever the type of tape reader, the switching effect is achieved by using binary arithmetic, a system that has 2 as a base and can convey numerical values in terms of 1 and 0: “on” and “off.”

To understand the binary system of numbers it is helpful to look more closely at the familiar decimal system, which uses 10 as a base. In this system nu-

merical values are constructed from multiples of units, tens, hundreds, thousands, etc.

Unit	$1 = 10^0$
Ten	$10 = 10^1$
Hundred	$100 = 10^2$
Thousand	$1000 = 10^3$

Thus the number 2345 is made up as follows:

Thousand	Hundred	Ten	Unit
(10^3)	(10^2)	(10^1)	(10^0)
2	3	4	5

This is a convenient, well-understood method of expressing numbers, but unfortunately it does not readily relate to the requirements of electrical switching control.

Now consider the application of the binary system using two as the base— 2^0 , 2^1 , 2^2 , 2^3 , and so on—and then relate this to certain decimal values as follows:

Let $2^0 = 1$

Then $2^1 = 2$

Then $2^2 = 4$

Then $2^3 = 8$ and so on

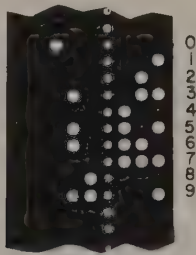
By using this small range of binary values any decimal digit can be expressed as shown in Table 5.1.

Now if the holes in the perforated control tape are arranged in columns or tracks corresponding to the binary values indicated in Table 5.1 it is possible

Table 5.1 Decimal digits expressed in binary.

Decimal Digit	Binary equivalent				Composition
	2^3	2^2	2^1	2^0	
1				1	$2^0 = 1$
2			1	0	$2^1 + 0 = 2$
3			1	1	$2^1 + 2^0 = 3$
4		1	0	0	$2^2 + 0 + 0 = 4$
5		1	0	1	$2^2 + 0 + 2^0 = 5$
6		1	1	0	$2^2 + 2^1 + 0 = 6$
7		1	1	1	$2^2 + 2^1 + 2^0 = 7$
8	1	0	0	0	$2^3 + 0 + 0 + 0 = 8$
9	1	0	0	1	$2^3 + 0 + 0 + 2^0 = 9$

to express the required decimal values by making perforations in the appropriate places:



The tape shown above has, eight tracks or vertical columns where a hole might be punched. Five columns are all that is required to express numbers, columns one through four indicate binary values and six indicates a zero. However, the numbers used in numerical control need an identity; for example, a slide movement not only has a dimensional value but the axis in which movement is required has to be defined. This definition is achieved by using letters, as explained in Chapter 1.

There are two tape standards in general use, the ISO (International Standards Organization or ASCII, American Standard Code for Information Interchange) and the EIA (Electrical Industries Association); the latter was developed in the United States of America and gained wide acceptance before the introduction of the ISO standards. The two tapes identify letters in different ways, but both use the binary coded decimal system for numbers. The following description is applicable only to the ISO standard.

The 26 letters of the alphabet are identified numerically from 1 to 26. We have seen that the digits 1 to 9 can be expressed using four binary columns. To include the numbers 10 to 26 requires a fifth column, a fifth track in the tape, so that the decimal value can be expressed *in one row* of punched holes as shown in Table 5.2.

Table 5.2 Letters of the alphabet expressed numerically.

Letter	Decimal Digit	Binary equivalent					Composition
		2 ⁴	2 ³	2 ²	2 ¹	2 ⁰	
J	10		1	0	1	0	2 ³ + 0 + 2 ¹ + 0 = 10
K	11		1	0	1	1	2 ³ + 0 + 2 ¹ + 2 ⁰ = 11
L	12		1	1	0	0	2 ³ + 2 ² + 0 + 0 = 12
M	13		1	1	0	1	2 ³ + 2 ² + 0 + 2 ⁰ = 13
N	14		1	1	1	0	2 ³ + 2 ² + 2 ¹ + 0 = 14
O	15		1	1	1	1	2 ³ + 2 ² + 2 ¹ + 2 ⁰ = 15
P	16	1	0	0	0	0	2 ⁴ + 0 + 0 + 0 + 0 = 16
Q	17	1	0	0	0	1	2 ⁴ + 0 + 0 + 0 + 2 ⁰ = 17
R	18	1	0	0	1	0	2 ⁴ + 0 + 0 + 2 ¹ + 0 = 18
S	19	1	0	0	1	1	2 ⁴ + 0 + 0 + 2 ¹ + 2 ⁰ = 19

There is, of course, a conflict as far as the first nine decimal digits are concerned. Does the value 7, for example, indicate a numerical value or the seventh letter, G? This is clarified by increasing the number of tracks in the tape from five to seven. Digits are indicated by additional holes being punched in both tracks five and six, while letters are indicated by holes punched in track 7.

The control system will require other characters as well as numbers and letters. For instance, a minus (-) sign may be necessary to indicate the direction of slide movement. These additional symbols have been allocated combinations of punched holes not used otherwise. Thus all the data required for CNC part programming purposes can be expressed via a seven-bit code. The term "bit," incidentally, is derived from BInary digiT.

Finally, to check the accuracy of the tape punching and tape reading, there is an eighth track referred to as a parity track. The ISO standard requires that each row contain an even number of holes. If the required character is expressed by an odd number of holes, an extra hole will be punched in track eight. If the required character is expressed by an even number of holes, there will be no extra hole in track eight. This system is referred to as "even parity." The EIA system also uses an eighth track, but as an end-of-block code. The fifth track is a parity punch to give odd parity in EIA-244-D standards. The newer EIA-358-B standard requires even parity and is becoming widely used in conjunction with computer applications. A visual check that each line of the tape contains an even number of holes (ISO) or an odd number of holes (EIA) is one method of ascertaining that there are no errors as a result of the equipment malfunctioning.

By reference to Figure 5.5 the reader can see the variations between the two tape standards. Most modern control systems will accept either standard EIA or ISO. (EIA standards 244 and 358 can be found in Appendix A.)

Tape Format Each horizontal row of holes in the tape is termed "character." Each set of characters is termed a "word." Each set of words is termed a "block." This is illustrated in Figure 5.6.

Blocks are identified by the letter N or O followed by three or four digits. A block can contain information on the type of slide movement required, length of slide movement, rate of slide movement, spindle speed, tool identity, etc., and will terminate with an "end of block" character.

The order in which words are entered in a block may be fixed or variable. The fixed block format requires each block to have the correct number of entries and they must appear in a set sequence. This means that data have to be reentered in each block even if there has been no change from the previous block and even if the numerical value of the entry is zero. On some systems each word has to be separated by the *tab* function, this type of format being referred to as *tab sequential*.

Much more commonly used is the variable block format in which words can be entered in any order. Their meaning or function is determined by the letter preceding the data, a system referred to as *word address*. Data that remain

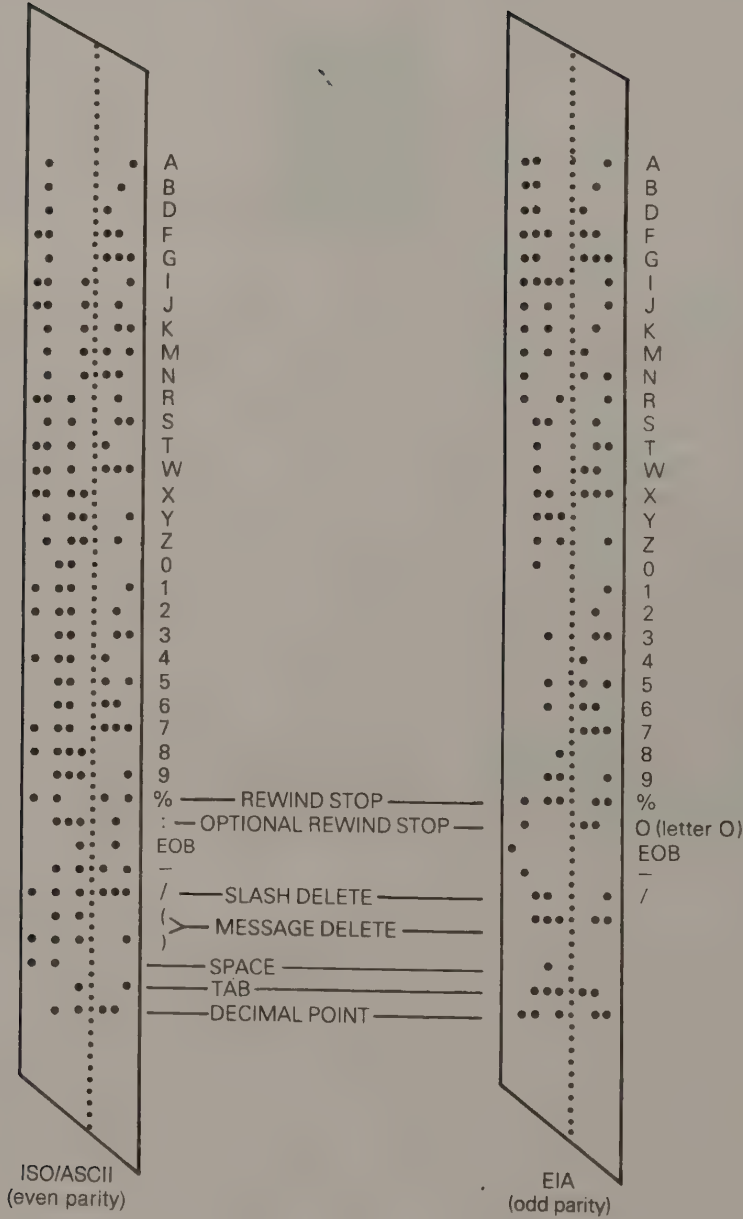


Figure 5.5 Standard tape codes.

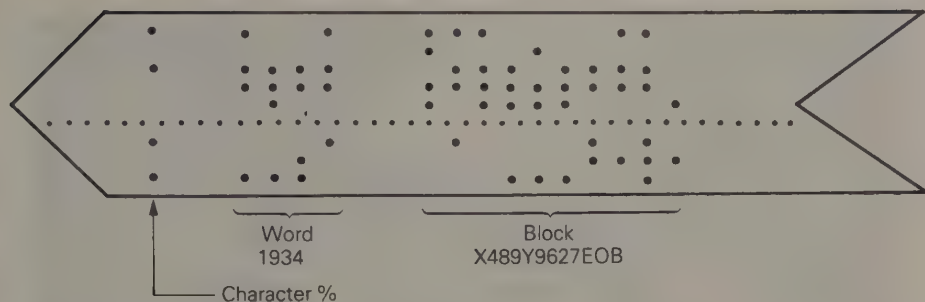


Figure 5.6 *Perforated tape format.*



Figure 5.7 *Automatic tape punch for interfacing with computer.*

unchanged in following blocks need not be reentered, and this leads to more rapid programming and a considerable reduction in the resulting tape length.

Production of Punched Tape Punched tapes may be produced either from a teletypewriter or by an automatic tape punch interfaced (connected) with a computer. An automatic tape punch is shown in Figure 5.7.

The teletypewriter is similar in many ways to an electric typewriter. It has an alphanumeric (letters and numbers) keyboard, and the prewritten program is typed in the normal way either onto conventional teletype paper or onto a blank program sheet. A teletypewriter is shown in Figure 5.8.

Attached to the side of the teletypewriter is the punching device. The blank tape feeds automatically from a roll into the punching head and, as each character is typed on the keyboard, a row of holes is punched in the tape.



Figure 5.8 Teletypewriter used for production of perforated tape.

Additional copies of the tape or of the typewritten program can be made automatically.

The facility to produce additional copies can be used for tape correcting. The original tape is run through the machine, and while it is running through, a replica of the tape and a printout are being made. At the point where the correction is to be made the automatic process is stopped. The new data are then typed into the program in the normal way. When the correction is completed the original tape is inched forward to the point where the original entry is still valid and the automatic reproduction process is restarted.

With computer-linked tape preparation facilities, editing is somewhat simpler. The listed part program appears on the display screen, where it can readily be examined, and any necessary alterations can be made via the keyboard.

When the programmer is satisfied that all is correct, the tape punch is activated to produce a complete new tape. Additional copies of the tape can readily be made and an interfaced printer will provide copies for filing for future reference.

Tape Proving Before a punched tape can be used for machining, it should be "proved," that is, checked that the desired machine movements will be achieved. This can be done on the machine tool, although the wisdom of wasting valuable machining time in nonproductive testing is questionable.

On basic numerically controlled machines, that is, those not computer controlled, testing facilities are limited to a "dry run," where all machine feedrates are changed to rapid transverse. Owing to the inherent danger present here from the flip of a switch, no part is placed in the machine and tooling may also be removed. It should be noted that a dry run only checks for command format errors, and after machine setup, the program should be cycled through in a block by block or single cycle mode to check for physical errors. On computer controlled machines the dry run can be complemented by a "test run," that is, all axis and spindle movements are inhibited, but the visual display is continually updated as demanded by the program in real time, that is, the actual time it would take to machine the component. Errors in the program, like no spindle speed stated would be indicated by an appropriate error message appearing on the CRT screen.

To avoid incurring the nonproductive downtime referred to, other test facilities remote from the machine may be more appropriate; for example, a plotter may be used. A plotter is in effect an automatic drawing device. The profile of the cutter path is traced out by the machine according to the data supplied via the tape. The result is, of course, a "flat" view; depth, that is, the third axis, is achieved by using colors and different views if available. An interfaced printer may be used to provide a copy of the program and a drawing of the tool path.

The tape-proving facilities referred to are largely redundant when computer graphics are used as part of the programming preparation process. The prepared program is fed into the computer via keyboard, floppy disk, or tape, the entry appearing on the CRT. Incorrect entries, for example, an unrealistic feed rate, can be stalled and the operator informed by a displayed error message. When the program is complete, it can be transferred to storage and the computer graphics are then used to simulate a test run. The correct blank size appears and, using animated tool movements, is "machined" according to the program requirements. The use of a computer for program proving is illustrated in Figure 5.9. A computer printout of a program, together with a graphical representation of the component, is shown in Figure 5.10. It should be noted that graphic types of prove-out do not eliminate the need for a slowed down single step mode run through the program after each setup.

When the programmer is satisfied that the entry is correct, a tape can be produced very rapidly at the touch of a key via an automatic tape punch in-

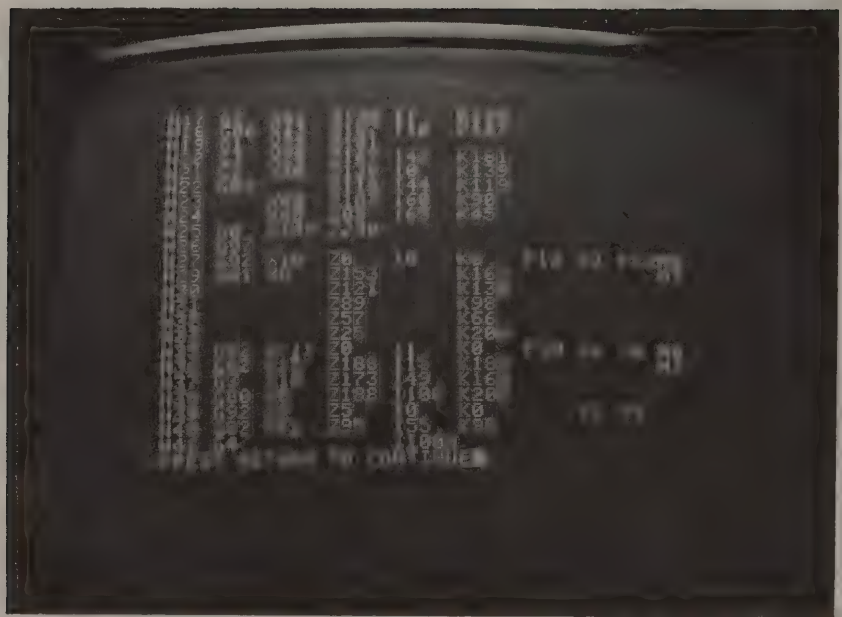


Figure 5.9 (a) Use of computer for program proving: program listing. (Metric output example for unidentified machine tool.)

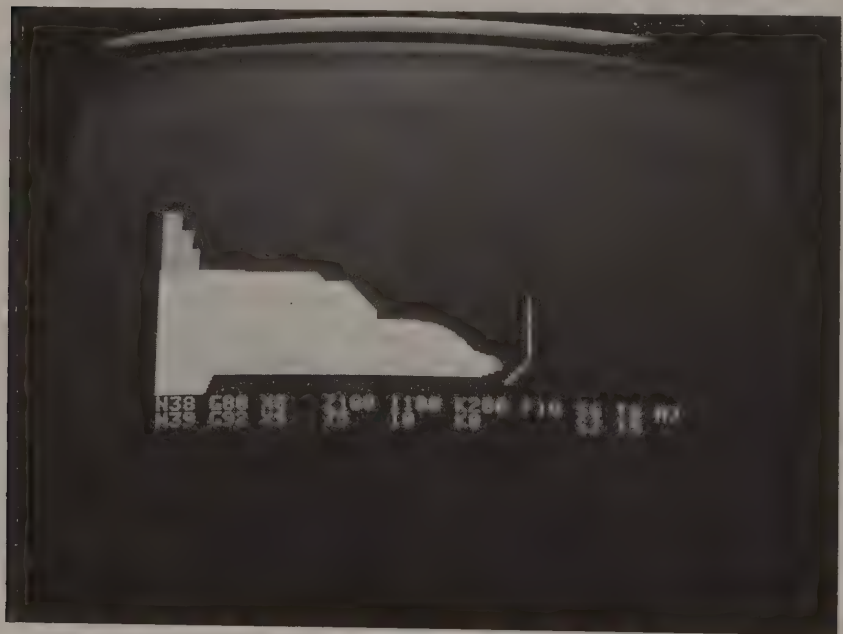


Figure 5.9 (b) Use of computer for program proving: graphical simulation.



Optional stop out.

Single step out.

Time Taken = 11.93 min.

=====

PROGRAM USED TO GENERATE WORKPIECE

=====

Figure 5.10 (a) Computer printout and graphical representation of component in metric output form for unidentified machine tool.

Figure 5.10 (b) Program used to generate workpiece.

N1	G96							S1000	
N2	G95								
N3	G92	X10	Z0	I0	K1	F99	S9	T1	M8
N4	G82	X61	Z188	I61	K91				M3
N5	G81	X68	Z84	I68	K188				M6
N6	G84	X56	Z188	I56	K109				M3
N7		X51		I51	K114				
N8		X46		I46	K119				
N9	G0	X41							
N10	G83		Z119	I45					
N11	G81	X63	Z101	I63	K188				
N12	G84	X36	Z188	I36	K159				
N13		X31		I31	K165				
N14		X27		I27	K170				
N15	G82	X23		I23	K181				
N16	G3	X29	Z167	I41					

N17	G2	X41	Z139	I1	K139				
N18	G80	X43	Z187	I16	K187				
N19	G1	X22	Z181						
N20	G3	X28	Z167	I42	K181				
N21	G2	X40	Z139	I0	K139				
N22	G83		Z119	I44	K119				
N23		X60	Z103	I60	K90				
N24		X65	Z85	I65	K45				
N25	G0	X145	Z230						
N26	G97								
N27	G92	X10	Z0	I0	K0	F12	S3	T12	M8
N29	G81	X0	Z153		K162				M3
N30			Z121		K130				
N31			Z89		K98				
N32			Z57		K66				
N33			Z25		K200				
N35	G92	X-11	Z0	I1	K0	F10	S6	T8	M8
N36	G82	X15	Z188	I15	K185				M3
N37	G3	X10	Z173	I49	K164				
N38	G80	X8	Z188	I100	K200				
N39	G92	X9	Z5	I0	K0		S2	T5	
N40	G82	X62	Z95	I55	K95				
N41	G4			I1000					
N42	G80	X68		I68	K40				
N43	G81	X60	Z40						
N44	G85	X68	Z36	I60	K36				
N45			Z31		K31				
N46			Z27		K27				
N47	G0		Z53						
N48	G1	X60	Z45						
N49	G80	X68	Z30	I100	K200				
N50	G92	X5	Z3	I0	K1	F99	S5	T6	
N51	G0	X66	Z100						
N52	G34	X64	Z37		K50				
N53		X63							
N54		X62							
N55	G0	X100	Z200						M2

terfaced with the computer. Similarly, an interfaced printer will produce a printout.

Magnetic Tape Data Input

Magnetic tape, in the form of cassettes, is a widely used means of transmitting data (Figure 5.11). The advantages claimed for it are:

- (a) easier handling;
- (b) more rapidly produced and read;
- (c) the program can be erased and the tape re-used;

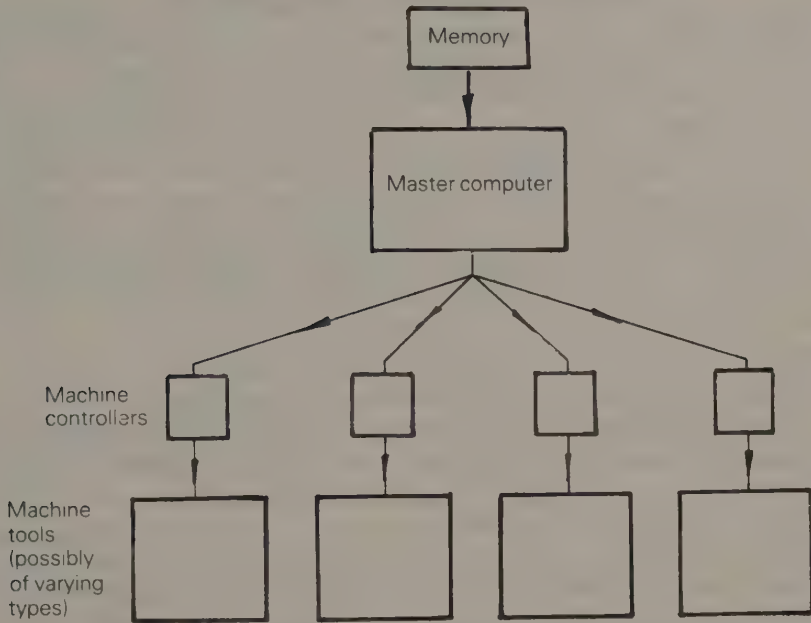


Figure 5.11 Data input from master computer: direct numerical control (DNC).

- (d) simpler editing;
- (e) more storage space than a paper tape of the same length;
- (f) more durable than paper tapes.

Early applications of magnetic tape involved a recording being made as machining of the first component was being carried out “manually” from the control console, a form of manual data input. A major drawback of this system was that the final program was only as rapid as the human reactions producing it and, with all the dial setting and switching involved, it was inevitably rather slow compared with modern techniques.

It was the advent of computer-controlled machining and computer-aided program proving that resulted in a more general application of magnetic tape. To record a part program in this way involves entering a program by MDI either at the machine control unit or through a computer keyboard. After the program has been entered it can be listed, edited and proved by using computer graphics, as discussed earlier. Finally, the program is recorded in much the same way as a tape recording of a piece of music can be made from a record or the radio.

Magnetic tape recordings do have the disadvantage of not being visible without the use of special CRT screens, printers, or plotters. Their other major disadvantage is that the magnetic information can be easily scrambled or destroyed if placed near magnetic fields or machine tools.

Data Input via Portable Electronic Storage Unit

There are problems associated with the use of paper and magnetic tape in what may be a dirty, and therefore potentially damaging, environment such as may be encountered in a machine shop. These problems may be eliminated by the use of a portable electronic storage unit. The data are first transferred into the unit away from the machine shop. The unit is then carried to the machine, connected to the machine control unit, and the data are then transferred. Similarly data already in the machine control unit may be transferred into the portable unit for downloading and storage elsewhere. Data transfer is very rapid. The capacity of the portable units is such that a number of programs can be accommodated at any one time.

Magnetic Disk Input via an Interfaced Computer

Providing the distance is not too great, it is possible to directly cable-link a microcomputer to a machine tool or, alternatively, to have a computer mounted on a trolley that can be brought alongside the machine and temporarily interfaced. It is then possible to transfer data stored on a magnetic diskette, commonly referred to as a "floppy disk," into the computer and hence into the machine control unit. Data already in the control unit can also be extracted and recorded. Newer CNC machine tool controls are now also appearing with portable or built in disk drives.

Disks have the same disadvantage as magnetic tape in that the program is not visible but, as with tapes, a program may be transferred into the computer for visual display, and printing and plotting facilities may be included.

The rate at which data can be transferred or retrieved using a disk is much faster than when using a tape and, size for size, the storage capacity of the disk is much greater.

Master Computer Data Input

An extension of the concept described previously of linking a computer to a machine tool is when a computer, usually a mini or mainframe, is permanently linked to a series of machines. Prepared programs stored in the memory of the master computer are then transferred to the microcomputer of the control unit of the selected machine tool as and when required. The concept is illustrated in Figure 5.11 and is referred to as direct numerical control (DNC).

Buffer Storage

Most modern control systems have a buffer storage, which is a capability to hold data extracted from the computer memory in an intermediate position. As one block of information is being processed the next is ready for instant transmission, the object being to speed the rate at which data are processed so that there is a minimum loss of machining time. Also the elimination of a dwell between blocks avoids marking the machined surface.

QUESTIONS

- 1 When would it be economically inadvisable to enter data manually into a machine control unit?
- 2 What is conversational manual input and what are the advantages and disadvantages of entering a machining program by this method?
- 3 Name three types of materials, or combinations of materials, used for perforated tape and state the advantages of each.
- 4 Name three types of tape readers. Which of these is most commonly used?
- 5 How is the reverse loading of perforated tape into a tape reader prevented?
- 6 Why is the binary system of numbers used to indicate the meaning of data input on a perforated tape?
- 7 What is the origin of the expression "eight bit" as applied to perforated tape?
- 8 Name the two tape standards in general use and explain how, by a visual check, you could identify them.
- 9 Explain what is meant by "character," "word," and "block" as applied to perforated tape format.
- 10 What is the difference between tab sequential and variable block tape format?
- 11 Describe two ways in which a tape can be proved away from the machine tool.
- 12 What is the difference between a dry run and a test run when checking data input?
- 13 What are the advantages of magnetic tape as a data storage medium?
- 14 What are the advantages of the floppy disk as a data storage medium?
- 15 What is a buffer storage and why is it necessary?

6

TERMS AND DEFINITIONS ASSOCIATED WITH PART PROGRAMMING AND MACHINE CONTROL

PART PROGRAMMING

The expression “part programming” causes some confusion, since “part” is often thought to mean something that is incomplete. In numerical control terms a part program is, in fact, a complete program. The word “part” means component.

PREPARATORY FUNCTIONS

Preparatory functions are used to inform the machine control unit of the facilities required for the machining that is to be carried out. For example, the control unit will need to know if the axis movements stated dimensionally in the program are to be made in inch or metric units, and whether the spindle is to rotate in a clockwise or counterclockwise direction.

The way in which machine controllers are provided with such information depends on the type of control unit. On conversational MDI systems, it may simply involve pressing the appropriate button on the control panel. For systems using the word address programming method, the various preparatory functions were originally standardized (ANSI/EIA RS274-D:1979; BS 3635:1972), each function being identified by the address letter G followed by two digits. Thus preparatory functions came to be referred to generally as “G codes.” The Standard has been adopted and is widely used, although variations in the allocation of special G codes will be encountered.

The preparatory functions, as they appear in the Standard, are shown in Table 6.1. The codes used for any particular control system will depend on the machine type and the sophistication of the system and, although a complete list such as the original standard is rather extensive, it should be appreciated that the number of codes included in any one system will be considerably fewer in number.

Table 6.1 Preparatory functions codes (M = modal).

Code Number	Function	Modal*
G00	Rapid positioning, point to point	(M)
G01	Linear positioning at controlled feed rate	(M)
G02	Circular interpolation CW—two dimensional	(M)
G03	Circular interpolation CCW—two dimensional	(M)
G04	Dwell for programmed duration	
G05	Unassigned EIA code may be used as hold. Cancelled by operator	
G06	Parabolic interpolation	(M)
G07	Unassigned EIA code reserved for future standardization	
G08	Programmed slide acceleration	
G09	Programmed slide deceleration	
G10 } G11 } G12 }	Unassigned EIA code sometimes used for machine lock and unlock devices	
G13–G16	Axis selection	(M)
G17	XY plane selection	(M)
G18	ZX plane selection	(M)
G19	YZ plane selection	(M)
G20	Unassigned EIA code	
G21 } G22 } G23 }	Unassigned EIA code sometimes used for nonstop blended interpolation movements	
G24	Unassigned EIA code	
G25–G29	Permanently unassigned. Available for indi- vidual use	
G30 } G31 } G32 }	Unassigned EIA code	
G33	Thread cutting, constant lead	(M)
G34	Thread cutting, increasing lead	(M)
G35	Thread cutting, decreasing lead	(M)
G36–G39	Permanently unassigned. Available for indi- vidual use	
G40	Cutter compensation/offset, cancel	(M)
G41	Cutter compensation, left	(M)
G42	Cutter compensation, right	(M)
G43	Cutter offset inside corner	(M)
G44	Cutter offset outside corner	(M)
G45 } G46 } G47 } G48 } G49 }	Unassigned EIA code	
G50	Reserved for adaptive control	
G51	Cutter compensation +/0	
G52	Cutter compensation -/0	
G53	Linear shift cancel	(M)
G54	Linear shift X	(M)

Table 6.1 (Continued)

Code Number	Function	Modal ^a
G55	Linear shift Y	(M)
G56	Linear shift Z	(M)
G57	Linear shift XY	(M)
G58	Linear shift XZ	(M)
G59	Linear shift YZ	(M)
G60–G69	Unassigned EIA codes	
G70	Inch programming	(M)
G71	Metric programming	(M)
G72	Circular interpolation—CW (three dimensional)	(M)
G73	Circular interpolation—CCW (three dimensional)	(M)
G74	Cancel multiquadrant circular interpolation	(M)
G75	Multiquadrant circular interpolation	(M)
G76–G79	Unassigned EIA code	
G80	Fixed cycle cancel	(M)
G81	Fixed cycle 1	(M)
G82	Fixed cycle 2	(M)
G83	Fixed cycle 3	(M)
G84	Fixed cycle 4	(M)
G85	Fixed cycle 5	(M)
G86	Fixed cycle 6	(M)
G87	Fixed cycle 7	(M)
G88	Fixed cycle 8	(M)
G89	Fixed cycle 9	(M)
G90	Absolute dimension input	(M)
G91	Incremental dimension input	(M)
G92	Preload registers	
G93	Inverse time feedrate (V/D)	(M)
G94	Inches (millimeters) per minute feedrate	(M)
G95	Inches (millimeters) per revolution feedrate	(M)
G96	Constant surface speed, feet (meters) per minute	(M)
G97	Revolutions per minute	(M)
G98 }	Unassigned EIA code	
G99 }		

^a Function retained until cancelled or superseded by subsequent command of same letter.

Many preparatory functions are modal, that is, they stay in operation until changed or cancelled.

MISCELLANEOUS FUNCTIONS

In addition to preparatory functions there are a number of other functions that are required from time to time throughout the machining program. For example, coolant may be required while metal cutting is actually under way but

will need to be turned off during a tool-changing sequence. Operations such as this are called “miscellaneous functions.”

Conversational MDI control systems will, as with preparatory functions, have their own particular way of initiating miscellaneous functions. but for word address systems the EIA standards have been adopted except for special options on particular machine tools. The functions are referred to as “M functions” and are identified by the address letter M followed by two digits.

The original standardized miscellaneous functions are listed in Table 6.2. The functions available will vary from one control system to another, the number available being fewer than the complete list.

POSITIONING CONTROL

The basis of numerically controlled machining is the programmed movement of the machine slides to predetermined positions. This positioning is described in three ways:

- (a) point-to-point
- (b) line motion or linear interpolation
- (c) contouring or circular interpolation.

Point-to-Point Positioning

Point-to-point positioning involves programming instructions that only identify the next position required. The position may be reached by movement in one or more axes. When more than one axis is involved, the movements are not coordinated with each other, even though they may occur simultaneously. The rate of movement is usually, although not necessarily, the maximum for the machine.

Figure 6.1 shows a component the machining of which would involve point-to-point positioning, the holes being drilled in the sequence A to D. Note that it is the positioning prior to drilling that is point-to-point, not the drilling operation itself.

Line Motion Control

Line motion control is also referred to as linear interpolation. The programmed movement results from instructions that specify the next required position and also the feed rate to be used to reach that position. This type of positioning would be involved in machining the slot in the component shown in Figure 6.2, the cutter moving in relation to the workpiece from point A to point B. Although a continuous cutter path appears to be the result, two distinct slide movements are involved, each slide movement being independent of the other.

Linear interpolation was initially defined as slide movement at programmed

Table 6.2 Miscellaneous functions codes.

Code number	Function	Function Starts Relative To Commanded Motion In Its Block		
		With	After Completion	Modal ^a
M00	Program stop		X	
M01	Optional stop		X	
M02	End of program		X	
M03	Spindle on CW	X		X
M04	Spindle on CCW	X		X
M05	Spindle off		X	X
M06	Tool change			
M07	Coolant 2 on	X		X
M08	Coolant 1 on	X		X
M09	Coolant off		X	X
M10	Clamp			X
M11	Unclamp			X
M12	Synchronization code		X	X
M13	Spindle on CW, coolant on	X		X
M14	Spindle on CCW, coolant on	X		X
M15	Motion in the positive direction	X		
M16	Motion in the negative direction	X		
M17 }	Unassigned EIA code. Reserved for future standardization			
M18 }				
M19	Oriented spindle stop			X
M20–M29	Permanently unassigned. Available for individual use			
M30	End of tape/data		X	X
M31	Interlock bypass	X		X
M32–M35	Unassigned EIA code			
M36 }	Permanently unassigned. Available for individual use			
M37 }				
M38 }				
M39 }				
M40–M46	Gear changes if used otherwise unassigned	X		X
M47	Return to program start			
M48	Cancel M49	X		X
M49	Feed/speed bypass override	X		X
M50–M57	Unassigned EIA code			
M58	Cancel M59	X		X
M59	Bypass constant surface speed updating	X		X
M60–M89	Unassigned EIA code			
M90–M99	Reserved for user			

^a Function retained until cancelled or superceded by subsequent command of same letter.

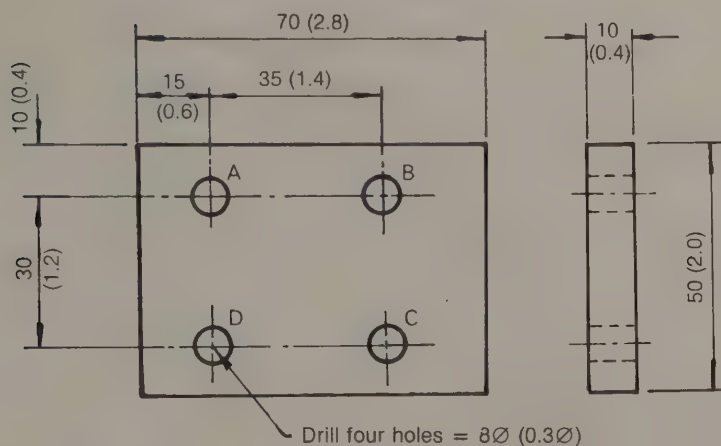


Figure 6.1 Component detail involving point-to-point positioning. (Inch units are given in parentheses.)

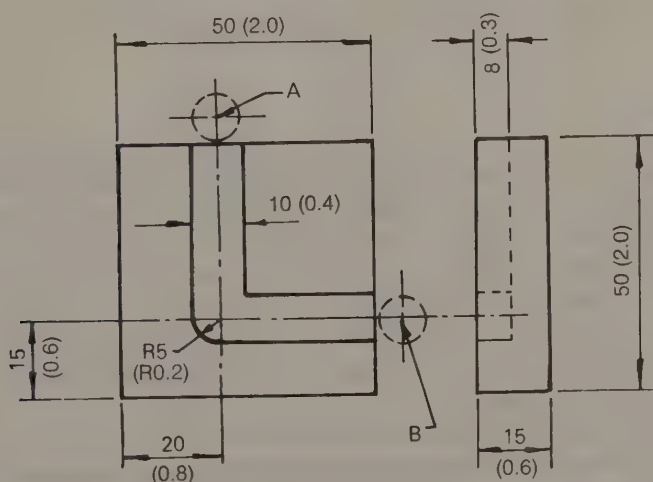


Figure 6.2 Component detail involving line motion control. (Inch units are given in parentheses.)

feed rates parallel to the machine axes. More recently it has also been used to describe linear movement when two, or sometimes three, slides are moving at the same time at programmed feed rates, a facility not available on earlier control systems. When two slides are moving simultaneously, an angular tool path results, and when three slides are involved, the result would be as indicated in Figure 6.3.

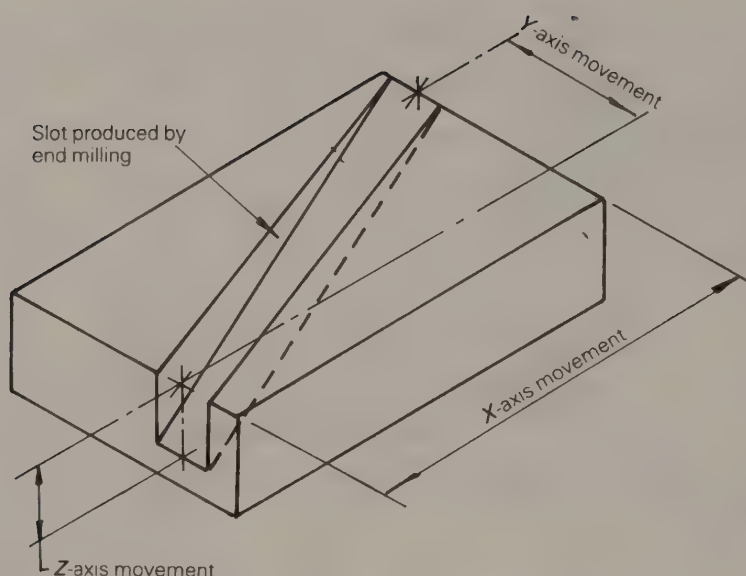


Figure 6.3 Resulting tool path when three slides move simultaneously.

Contouring

Contouring also involves two or more controlled slide movements resulting from program data that specify the next position required and the required feed rates to reach that position, so there is some overlap between linear interpolation and contouring. However, contouring can also be much more complex, involving combinations of angular movement and curves with one feature moving without interruption in the cutting process into another. This type of movement gives rise to the expression continuous path machining, which is often used to describe contouring.

Machining of the elliptical profile shown in Figure 6.4 would involve continuous path movement. Likewise, the radii shown on the components in Figure 6.5 would be produced in a similar manner. The elliptical shape is not readily defined in numerical terms, and to produce the necessary cutter path would present an interesting, although not insurmountable, problem to the part programmer unless the control system was specially equipped with a canned cycle to deal with such a situation. On the other hand, the two radii shown in Figure 6.5 are an everyday occurrence and most control systems can readily accommodate the production of a radius, or a combination of radii. Such a facility is referred to as circular interpolation.

Circular arcs may be programmed in the XY , XZ , and YZ planes. In exceptional cases three axes may be involved, resulting, in effect, in a helical tool path.

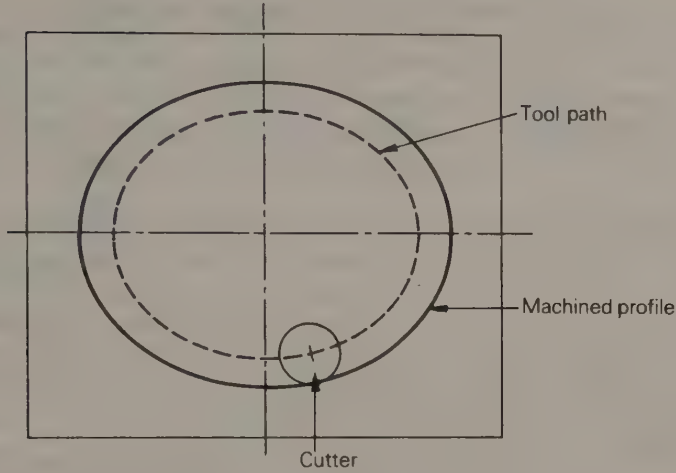


Figure 6.4 Component profile produced by contouring.

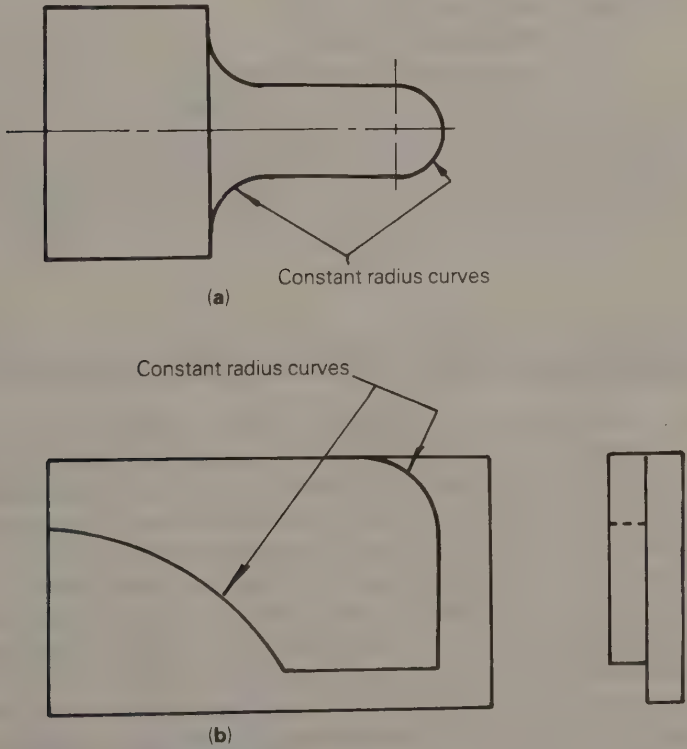


Figure 6.5 Components with radial features requiring circular interpolation: (a) turned component and (b) milled profile.

PROGRAMMING POSITIONAL MOVES

In practice the three types of positioning referred to previously are rarely isolated. The production of the majority of components will involve a combination of the techniques. However, it will be necessary to clearly identify in the part program the type of positioning required at each stage of the machining process.

Manual data input systems will vary from one control system to another. For example, a widely used training machine specifies all linear movement as linear interpolation and differentiates by linking the movement to an appropriate feed rate. The program entry is reduced to pressing a linear interpolation key followed by the dimensional detail and the feed rate. Similarly, a radius is simply defined by pressing a circular interpolation key, followed by a data entry of the dimensional value of the target position, the radius, and the direction of rotation as either clockwise or counterclockwise.

Control systems using the recommendations contained in EIA RS-274-D or BS 3635:1972 will specify the type of positioning involved by using the appropriate preparatory function or G code, the common ones being as follows:

- G00 Point-to-point
- G01 Linear interpolation
- G02 Circular interpolation clockwise
- G03 Circular interpolation counterclockwise

Having defined the type of positioning in this way the instruction is completed by including dimensional details of the move together with the feed rate for G01, G02, and G03. G00 moves are usually made at the maximum slide traverse rate for the machine.

DIMENSIONAL DEFINITIONS OF SLIDE MOVEMENT

In Chapter 1 it was explained that the axes in which slide movement can take place are designated by a letter and either a plus (+) or minus (-) sign to indicate the direction of movement. Unfortunately, these designated slide movements, owing to the different design configurations of machine tools, do not always coincide with the movement of the tool in relation to the work, and as a result this can cause some confusion when slide movements are being determined. In the case of a turning center with a conventional tool post there is no problem, since the slide movement and the tool movement in relation to the work are identical. But on a vertical machining center, for example, to achieve a positive (+) movement of the tool in relation to the work, the table, not the cutter, has to move, and this movement is in the opposite direction. Since a move in the wrong direction, especially at a rapid feed rate, could have disastrous results, this fact should be clearly understood.

A sound technique when determining slide movements is to program the tool movement in relation to the work. In other words, on all types of machines, imagine it is the tool moving and not, as is sometimes the case, the workpiece. To do this it is necessary to redefine some, but not all, of the machine movements. A simple diagram such as the one alongside the components shown in Figures 6.6 and 6.9 is usually very helpful.

Once the direction of movement has been established it will need to be dimensionally defined. There are two methods used, and they are referred to as:

- (a) absolute;
- (b) incremental.

Figure 6.6 shows the profile of a component to be machined on a turning center using the machine spindle center line and the face of the workpiece as datums in the X and Z axes respectively. Assume the sequence of machining is to commence with the 1.4 in. (35 mm) diameter, followed by the 1.2 in. (30 mm) diameter and finishing with the 1 in. (25 mm) diameter.

To machine the profile using absolute dimensions, it is necessary to relate all the slide movements to a preestablished datum. The movements required in absolute terms are indicated in Figure 6.7. Note that all position commands are the actual distance that the tool tip is from the datum point.

Incremental positioning involves relating the slide movement to the final position of the previous move. The slide movements, expressed in incremental terms, which would be necessary to machine the profile are indicated in Figure 6.8. Note position commands indicate the direction and the exact amount of slide motion required.

Note that each dimension in the X axis in Figure 6.7 is equal to the work

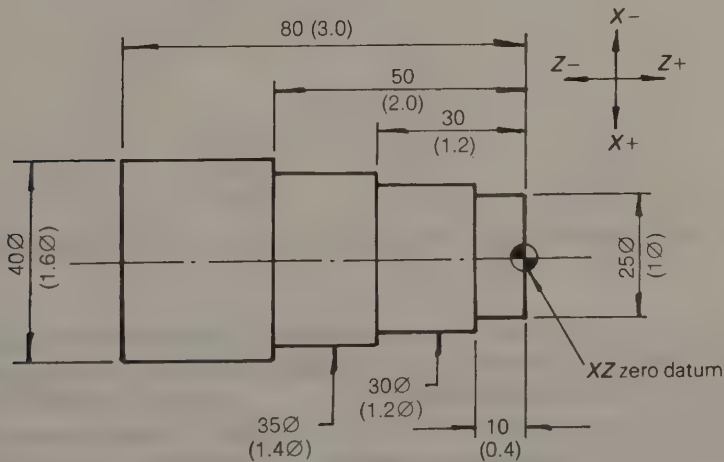


Figure 6.6 Component detail. (Inch units are given in parentheses.)

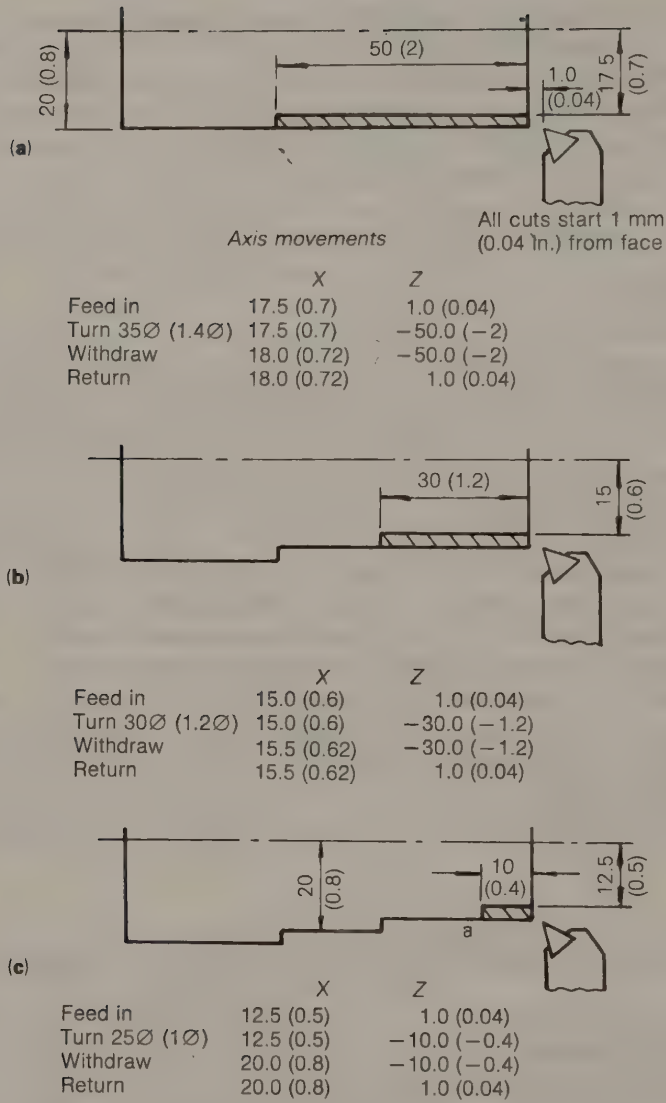


Figure 6.7 Turning using absolute positioning. (Inch units are given in parentheses.)

radius. When turning, some control systems will require dimensions in the X axis to be stated as a diameter, other machines may allow the programmer to select radius or diameter programming.

Figure 6.9 shows a component that is to be milled in the sequence A to C on a vertical machining center using datums as indicated. Assume that the movement in the Z axis to give a slot depth of 0.4 in (10 mm) has already

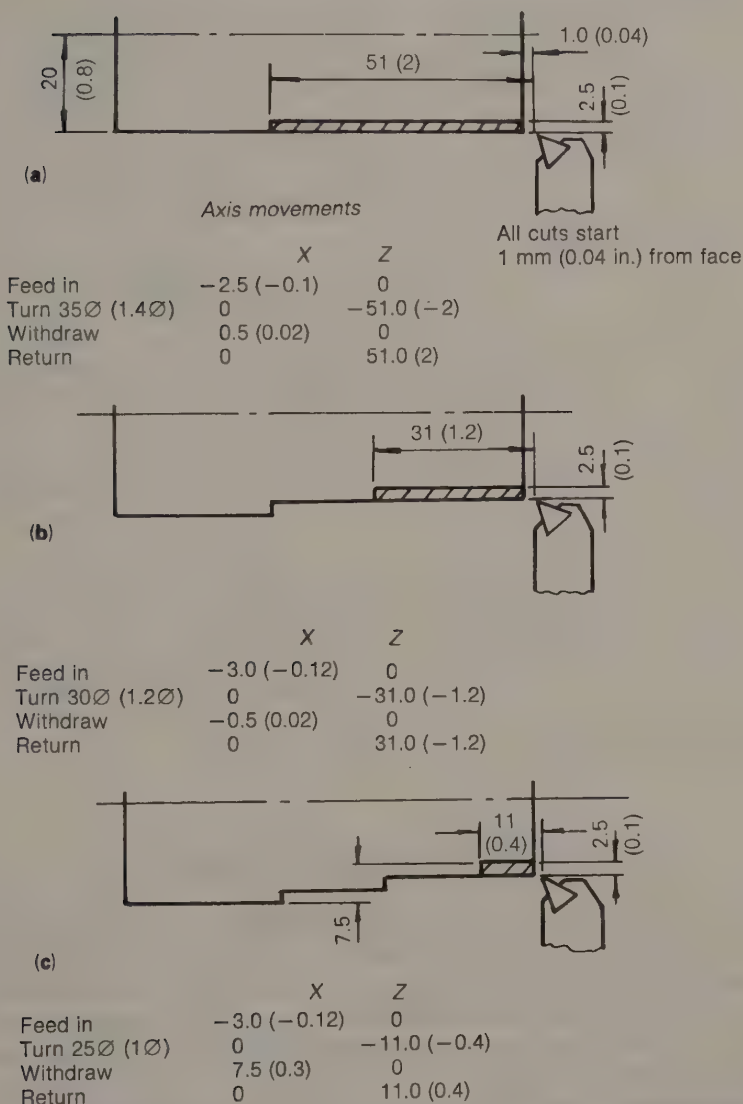


Figure 6.8 Turning using incremental positioning. (Inch units are given in parentheses.)

been made. The necessary slide movements in the X and Y axes in absolute and incremental terms are indicated in Figures 6.10 and 6.11, respectively.

On the more sophisticated control systems, it is possible to use absolute and incremental dimensional definition within the same program, the distinction being achieved by using the G91 preparatory function code when the switch from absolute (G90) to incremental (G91) is to be made.

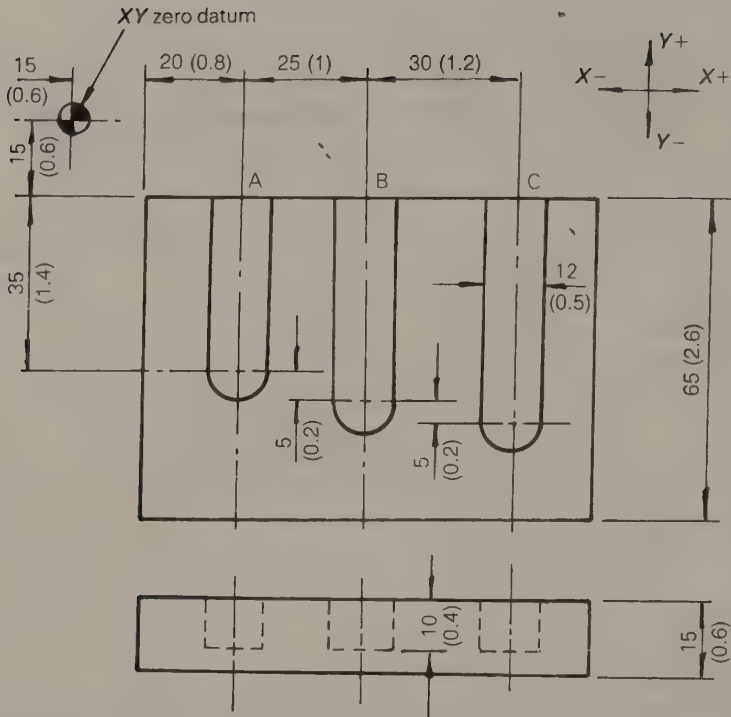


Figure 6.9 Component detail. (Inch units given in parentheses.)

CIRCULAR INTERPOLATION

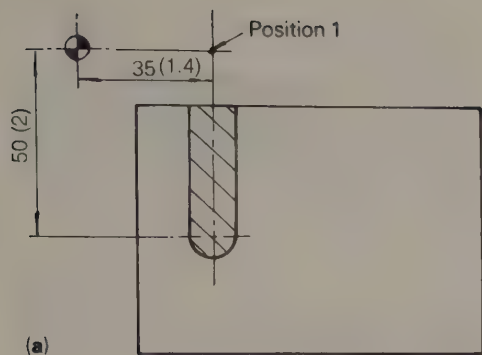
It was stated earlier that circular arc programming, particularly on conversational data input systems, has been reduced to simply dimensionally defining the target position, the radius, and the direction in which movement is to take place. On control systems using the word address format, it is rather more complex and there are slight variations in approach. Two of these variations will be considered later.

Common to all systems used to program circular movement is the need to determine whether the relative tool travel is in a clockwise (CW) or counter-clockwise (CCW) direction. The following approach is usually helpful.

1. For milling operations look along the machine spindle toward the surface being machined.
2. For turning operations look on to the top face of the cutting tool. (For inverted tooling this involves looking at the tool from below.)

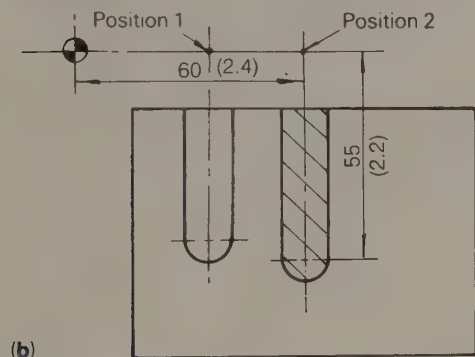
The standard G codes for circular interpolation are G02 (CW) and G03 (CCW). However, not all systems adopt this recommendation and there is at least one widely used system in which they are reversed, that is, G02 is CCW and G03

Figure 6.10 Milling using absolute positioning. (Inch units are given in parentheses.)

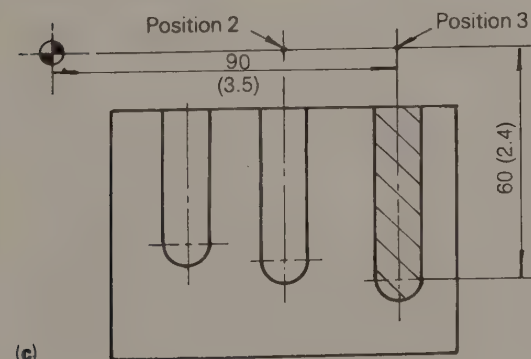


Axis movements

	X	Y
Move from datum to position 1	35.00 (1.4)	0
Mill to length	35.00 (1.4)	-50.00 (-2.00)
Return to position 1	35.00 (1.4)	0

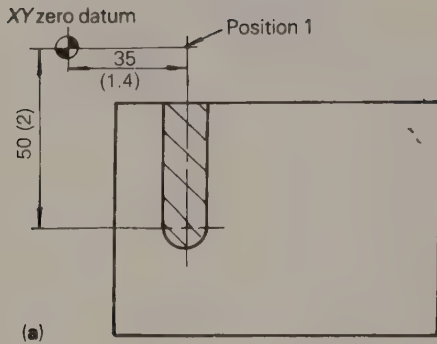


	X	Y
Move from position 1 to position 2	60.00 (2.4)	0
Mill to length	60.00 (2.4)	-55.00 (-2.2)
Return to position 2	60.00 (2.4)	0



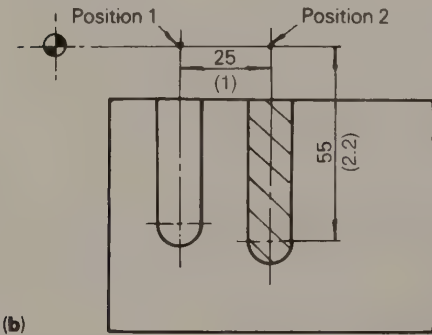
	X	Y
Move from position 2 to position 3	90.00 (3.5)	0
Mill to length	90.00 (3.5)	-60.00 (-2.4)
Return to position 3	90.00 (3.5)	0
Return to datum	0	0

Figure 6.11 Milling using incremental positioning. (Inch units are given in parentheses.)

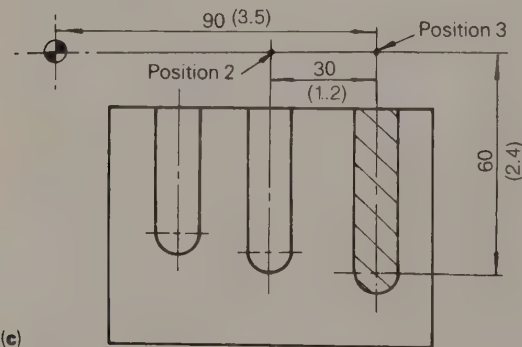


Axis movements

	X	Y
Move from datum to position 1	35.00 (1.4)	0
Mill to length	0	-50.00 (-2.00)
Return to position 1	0	50.00 (2.00)



	X	Y
Move from position 1 to position 2	25.00 (1.00)	0
Mill to length	0	-55.00 (-2.2)
Return to position 2	0	55.00 (2.2)



	X	Y
Move from position 2 to position 3	30.00 (1.2)	0
Mill to length	0	-60.00 (-2.4)
Return to position 3	0	60.00 (2.4)
Return to datum	-90.00 (-3.5)	0

is CW. (In this case it is advisable to refer to the machine tool programming manual.)

The three variations in arc programming referred to above are as follows. Note: That machines will normally not have all three methods of circular arc programming.

Method 1

Assuming that the last programmed move brought the cutting tool to the start point, the arc is defined in the following manner:

1. The finish or target point of the arc is dimensionally defined in relation to the start point using the appropriate combination of X, Y, and Z dimensional values stated in absolute or incremental terms.
2. The center of the arc is dimensionally defined in relation to the start point using I, J, and K values measured along the corresponding X, Y, and Z axes respectively.

Thus the arc shown in Figure 6.12 would be programmed as follows. In absolute terms using diameter programming:

<i>Inch</i>	G02	X	Z	I	K
		1.6	2.0	0	0.8
<i>Metric</i>	G02	X	Z	I	K
		40	50	0	20

In incremental terms:

<i>Inch</i>	G02	X	Z	I	K
		0.8	-0.8	0	0.8
<i>Metric</i>	G02	X	Z	I	K
		20	-20	0	20

The variation in the X values in these two examples is because the absolute program assumes that X values are programmed as a diameter rather than a radius.

I has no value because the center and start point of the arc are in line with each other in relationship to the X axis. In practice, when a value is zero, it is not entered in the program.

The I, J, and K values are always positive, with I related to X, J related to Y, and K related to Z.

Complete circles and semicircles are programmed as a series of 90° quadrants in many cases. Thus a complete circle would require four lines of program entry. New pieces of equipment can now complete full circles in one line of program entry.

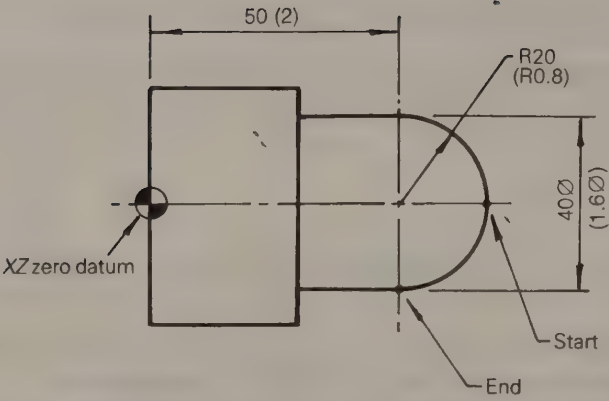


Figure 6.12 Turned component detail involving arc programming. (Inch units are given in parentheses.)

Figure 6.13 shows the program for a milled profile. The cutter radius has been ignored.

In absolute terms:

Inch	G	X	Y	I	J
	03	2	-1.2	1.2	0
	02	3.5	-1.2	0	1.6
Metric	G	X	Y	I	J
	03	50	-30	30	0
	02	90	-70	0	40

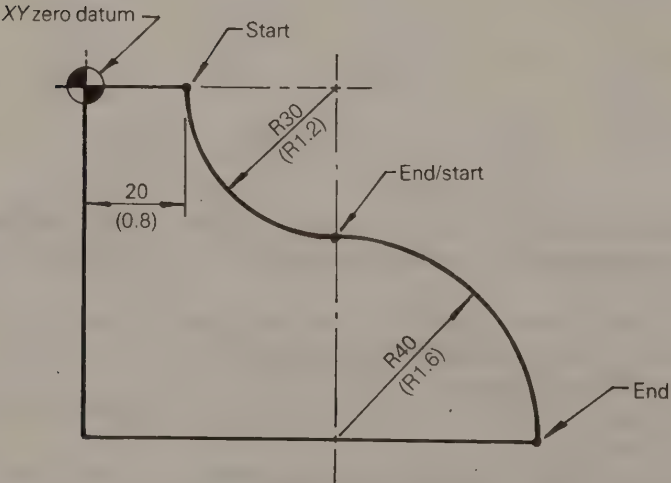


Figure 6.13 Milled component involving arc programming. (Inch units are given in parentheses.)

In incremental terms:

<i>Inch</i>	G	X	Y	I	J
	03	1.2	-1.2	1.2	0
	02	1.6	-1.6	0	1.6
<i>Metric</i>	G	X	Y	I	J
	03	30	-30	30	0
	02	40	-40	0	40

There are often situations where the start and/or stop points do not coincide with an X, Y, or Z axis, and it is then necessary to make a series of calculations. Such a situation is shown in Fig. 6.14. Dimensional values for X, Y, I, and J have to be determined. The necessary trigonometry is indicated in Fig. 6.15.

From A to B the magnitude of the X move is

$$\text{Inch } 1 \times \cos 30^\circ - 1 \times \cos 75^\circ = 0.866 - 0.259 = 0.607$$

$$\text{Metric } 25.00 \cos 30^\circ - 25.00 \cos 75^\circ = 21.65 - 6.47 = 15.18$$

From A to B the magnitude of the Y move is

$$\text{Inch } 1 \times \sin 75^\circ - 1 \times \sin 30^\circ = 0.966 - 0.500 = 0.466$$

$$\text{Metric } 25.00 \sin 75^\circ - 25 \sin 30^\circ = 24.15 - 12.50 = 11.65$$

The magnitude of the I dimension in the X axis is

$$\text{Inch } 1 \times \cos 75^\circ = 0.259$$

$$\text{Metric } 25 \cos 75^\circ = 6.47$$

The magnitude of J in the Y axis is

$$\text{Inch } 1 \times \sin 75^\circ = 0.966$$

$$\text{Metric } 25.00 \sin 75^\circ = 24.15$$

Once the dimensions have been incorporated, they are incorporated in the program as before.

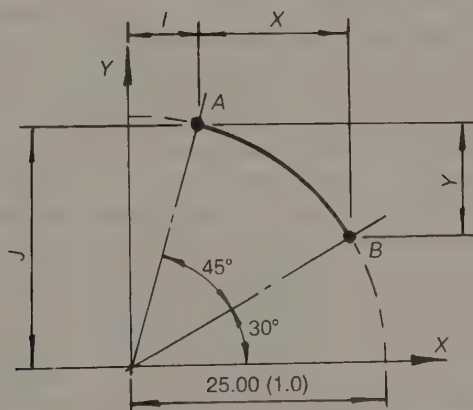


Figure 6.14 Partial arc programming. (Inch units are given in parentheses.)

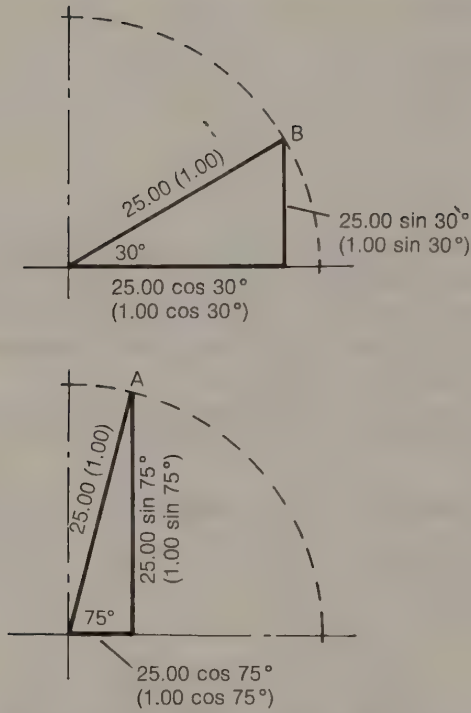


Figure 6.15 Trigonometry required to program a partial arc. (Inch units are given in parentheses.)

Method 2

The second method of arc programming varies from the one previously described in the way in which the arc center is defined. As in the previous method, it will be assumed that the cutting tool has arrived at the start point of the curve. To continue, the following data are required.

- 1. The finish or target point of the arc is dimensionally defined in relation to the start point using the appropriate combination of X, Y, and Z values stated in absolute or incremental terms.
- 2. The center of the arc is dimensionally defined in relation to the program datum using I, J, and K values measured along the corresponding X, Y, and Z axes respectively.

Using this method the arc shown in Fig. 6.12 would be programmed as follows.
In absolute terms:

Inch	G	X	Z	I	K
	02	1.6	2.0	0	2.0
Metric	02	40	50	0	50

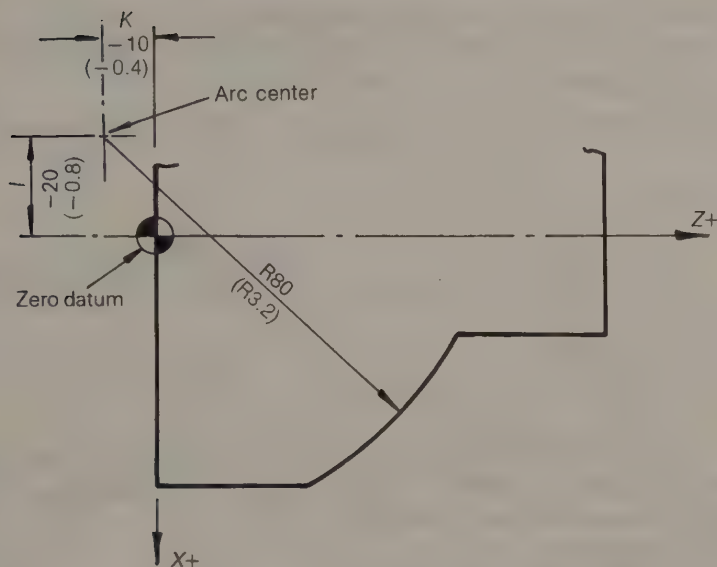


Figure 6.16 Negative *I* and *K* values. (Inch units are given in parentheses.)

In incremental terms:

Inch	G	X	Z	I	K
	02	0.8	-0.8	0	2.0
Metric	G	X	Z	I	K
	02	20	-20	0	50

Note that in this example it is *I* that has no value, since the center of the arc lies on the *X* datum and therefore *I* would be omitted from the program.

When the arc center is related to the program datum it is possible for the *I*, *J*, and *K* values to be a negative quantity, as illustrated in Figure 6.16.

The programming methods referred to above concern arcs of up to 90°. Some of the more modern control systems permit programming of arcs in excess of 90° in one data block, a facility referred to as 'multi-quadrant' programming or 360° circular interpolation.

Method 3

The third method of arc programming on some controls is to use absolute or incremental polar coordinates. It varies from the previous methods in that it does not use *I* and *J* values. With this method the circle center point has been defined previously with *X*, *Y*, or *Z* values. The arc is then programmed with a radius dimension and an angular amount of tool path from the circle center. A positive or negative angle will establish the direction of the cutter path:

1. The circle center is established with absolute or incremental dimensions.

2. The tool will have been moved to the arc start point.
3. The degrees of arc and radius are then programmed, with the sign (+ or -) on the degrees of arc establishing direction of cut.

Using the above terms the arc in Figure 6.12 would be programmed as follows:

In absolute terms:

<i>Inch</i>	CC X0 Z2 G1 X0 Z2.8 C Polar radius 0.8 polar angle 90°	Define circle center Position cutter to starting point Cut circle
<i>Metric</i>	CC X0 Z50 G1 X0 Z70 C Polar radius 20 polar angle 90°	Define circle center Position cutter to start point Cut circle

Positive angles denote clockwise motion; negative angles denote counterclockwise motion. Note: Most machines with polar coordinate circular interpolation capabilities are conversationally controlled.

In incremental terms:

<i>Inch</i>	CC X0 Z2 G1 X0 Z2.8 C Polar radius 0.8 polar angle 90°	Circle center from datum Position tool to start from circle center Circular movement
<i>Metric</i>	CC X0 Z50 G1 X0 Z20 C Polar radius 20 polar angle 90°	Circle center from datum Position tool to start from circle center Circular movement

RAMP

The starting and stopping of slide servo motors appear to be instantaneous. In fact there is, of course, a brief period of acceleration at the start of a move and a brief period of deceleration at the end of a move. This is shown graphically in Figure 6.17.

The period of acceleration is known as "ramp up" and the period of deceleration as "ramp down." The ramp is a carefully designed feature of the servo motor.

From a metal-cutting point of view, the quicker a slide attains its correct feed rate the better, and ideally this should be maintained throughout the cut. The ramp period therefore is kept as brief as possible, but consideration has to be given to ensuring that at the end of the movement there is no motor overrun or oscillation, both of which could affect the dimensional accuracy of the component.

For linear interpolation the ramp effect is rarely of concern, but for circular

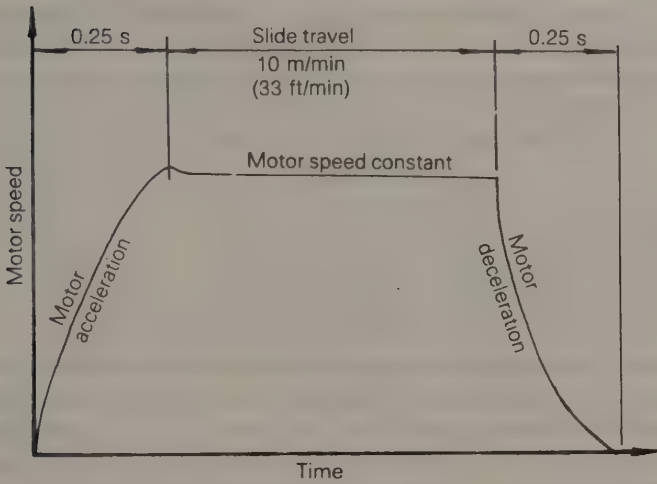


Figure 6.17 Servo motor speed/feed rate relationship.

interpolation, and particularly where one curve runs into another, it is preferable that there is no speed variation of the servo motor, and thus of the feed rate of the slide, however small this might be. Any such variation would not only affect the metal-removal rate but may also affect the dimensional accuracy and surface finish of the component. Because of this, many control units are equipped with a *ramp inhibit* or *ramp suppression* facility, which means there is no slowing down or acceleration of the slide movement as one programmed movement leads into a second. G codes allocated to ramp are usually G08 and G09.

REPETITIVE MACHINING SEQUENCES

There are a number of machining sequences that are commonly used when machining a variety of components. Other less common sequences may be repetitive, but only on one particular component. It is helpful, since it reduces the program length, if such a sequence can be programmed just once and given an identity so that it can be called back into the main program as and when required. Such sequences are referred to in a variety of ways, for example, as cycles, subroutines, loops, patterns, and macros. Although this can be slightly confusing, there are instances when one particular title appears to be more appropriate than the others. Various types of repeat machining sequences are discussed here.

Standardized Fixed Cycles

A number of the basic machining sequences, or cycles, commonly used were initially standardized (ANSI/EIA RS-274-D:1979; BS 3635:1972). The rec-

ommendations were commonly adopted and continue to be employed today. The machining cycles are identified by assigned G codes, and when they are incorporated into a control system, they are referred to as “fixed” or “canned” cycles. Perhaps the most commonly used fixed cycle is that of drilling a hole. Consider the hole shown in Figure 6.18(a). The sequence of machine movements involved in drilling the hole would be:

1. Position to hole location.
2. Lower the spindle at a programmed feed rate.
3. Lift the spindle rapidly to the start position.

Now consider the process of drilling the hole shown in Figure 6.18(b). The same sequence of spindle movements is necessary; the only variation is in the depth of travel. To program such a sequence of moves is quite simple, but if there were a large number of holes to be drilled, apart from the boredom of repeating the necessary data when writing the program, the program itself would be very long. In addition, the fewer data commands that have to be handled the less likely it is that errors will be made. By standardizing the sequence of moves the only additional data requirements are the new hole location, depth of cut, feed rate, and spindle speed. This information, with the appropriate G code, is entered only once. Each time the slide moves to bring the spindle to a new position in relation to the work another hole is drilled to the programmed depth.

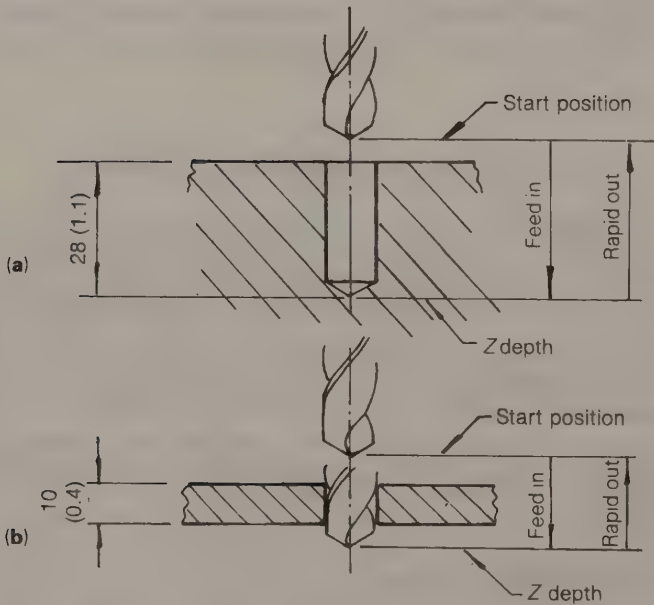


Figure 6.18 Movements required to drill holes. (Inch units are given in parentheses.)

Nonstandardized Fixed Cycles

It is often the case that manufacturers of machine control units wish to include in their systems cycles that are not necessarily widely applicable and therefore do not fit into the “standardized” category, but the inclusion of which considerably enhances their control system. The cycles they choose to include will depend on the machine type to which the control is to be fitted. Some of the more common cycles of this nature are discussed below.

Loops The term “loop” is particularly relevant when reducing raw material to size by making a series of roughing cuts. Consider the component shown in Figure 6.19, which is to be reduced from 50 mm (2 in.) to 26 mm (1 in.) diameter by a series of cuts each of 2 mm (0.08 in.) depth. Assuming that the starting point for the tool is as shown, the tool will first move in a distance of 2.5 mm (0.1 in.), thus taking a 2 mm (0.08 in.) depth of cut, travel along a length of 50 mm (2 in.), retract 0.5 mm (0.02 in.), and return to the Z datum, thereby completing the loop. It will then move in a distance of 2.5 mm (0.1 in.), feed along 50 mm (2 in.), retract 0.5 mm (0.02 in.), and return to the Z datum, and so on. The loop, including the feed rate, is programmed just once, but is repeated via the “loop count” command in the main program as many

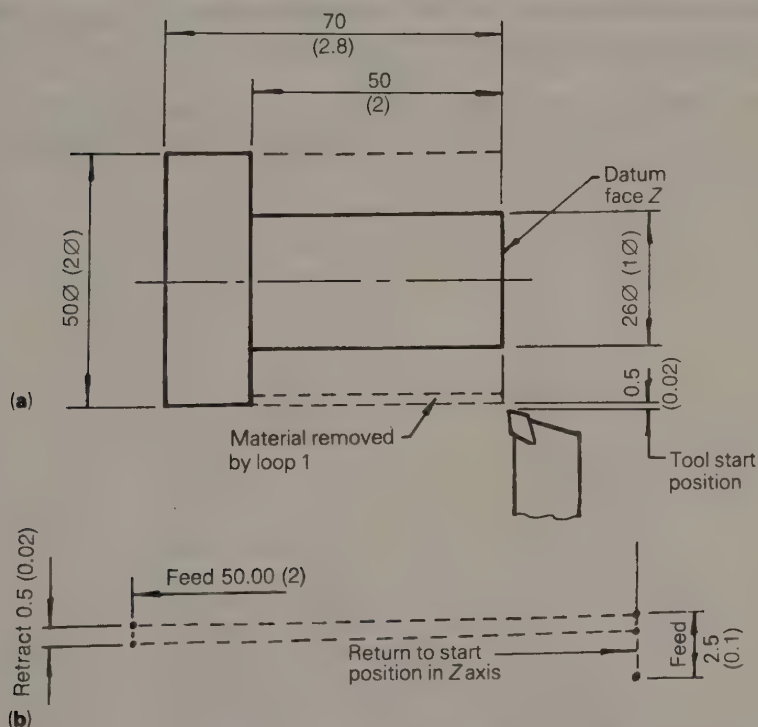


Figure 6.19 Looping or roughing cycle: (a) component and (b) loop details, repeated six times. (Inch units are given in parentheses.)

times as necessary to reduce the work to the required diameter. Note: some controls will do this with special "G" codes, while other controls will use special command codes, but the results are the same.

Face Milling Cycle Figure 6.20 shows details of a face milling cycle. After programming the appropriate G code, together with spindle speed and feed rate, the only other information required are the X and Y dimensions of the face to be milled. The control unit computer will determine the number of passes necessary and the appropriate cutter step-over to machine the face. The cutter diameter will be picked up automatically from previously entered information. This type of cycle is very commonly found on conversationally programmed controls.

Slot Milling Cycle Figure 6.21 illustrates a slot milling routine. As with face milling, the programmer has to state spindle speed, feed rate, and slot dimensions in the X and Y axes. The first pass made by the cutter passes through the middle of the slot and then returns to the start. Further passes are made until the correct depth is achieved, the number of passes necessary being determined by the axis increment depth programmed in the cycle. When the correct depth is reached, the cutter path is that of a series of cycles increasing in size with each pass. Some controls vary this process by cutting the entire slot at each depth except the finish pass. Again, as with the face milling, the computer will determine the step-over and the number of cycles necessary to machine the slot to size.

Pocket Milling Figure 6.22 illustrates the pocket milling cycle. This cycle starts at the center of the pocket, the cutter feeding in the Z axis to a pro-

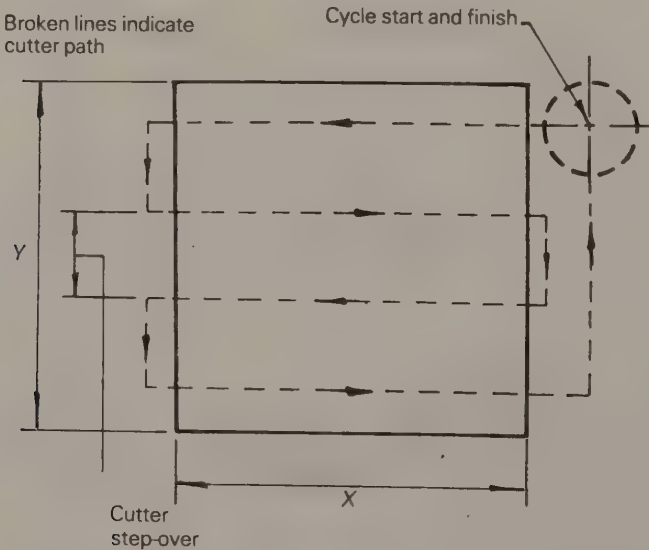


Figure 6.20 Face milling cycle.

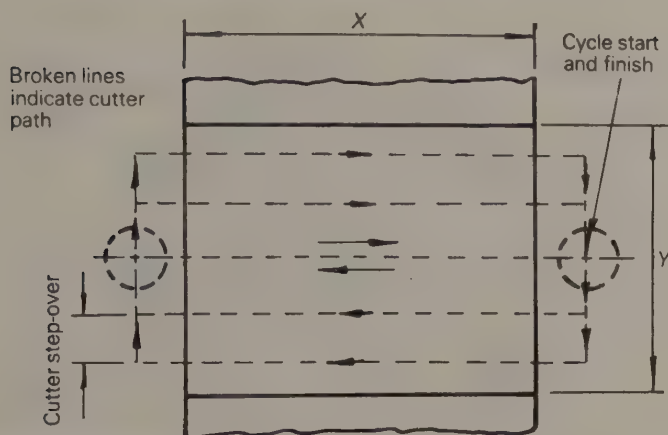


Figure 6.21 Slot milling cycle.

grammed depth. There follows a series of cycles until the programmed X and Y dimensions are reached, the step-over of up to 80% of the cutter diameter will ensure that a flat surface is produced by providing overlap of passes. Some systems provide for a cycle that roughs out the main pocket and then machines to size with a small finishing cut. If the pocket depth is such that more than one increment in the Z axis is necessary, the slide movement returns the cutter to the center of the pocket and the cycle is repeated at the next depth.

Bolt Hole Circles The term “bolt hole circle” means that a number of holes are required equally spaced on a stated pitch circle diameter as illustrated in Figure 6.23. Given that the program has brought the cutter to the pole position,

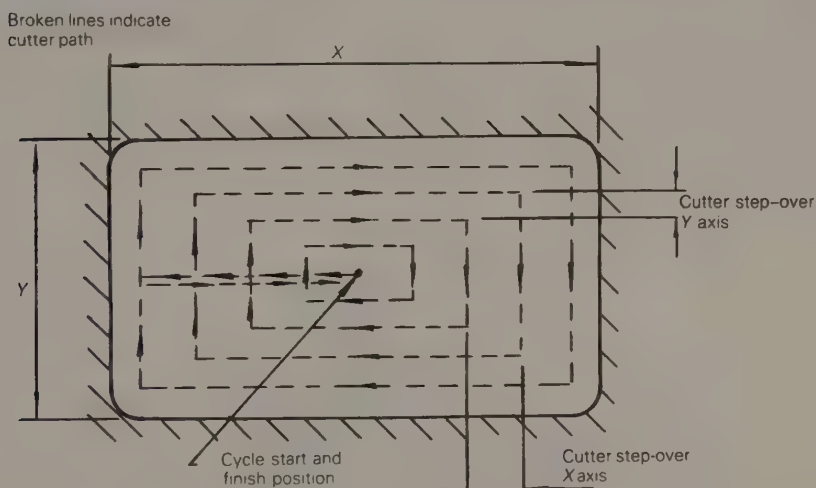


Figure 6.22 Pocket milling cycle.

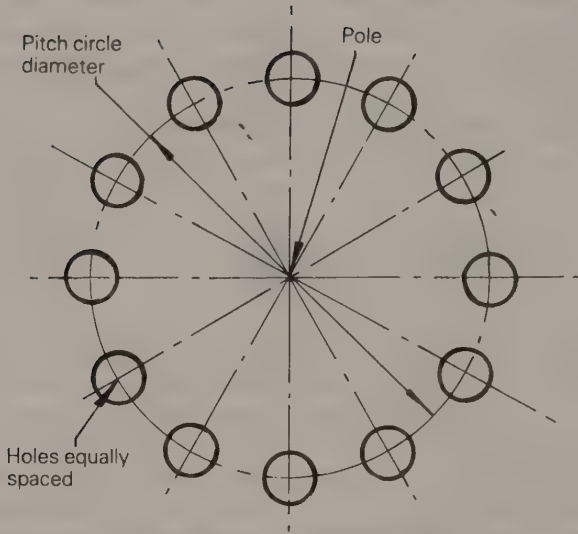


Figure 6.23 Bolt hole circle.

the other dimensional data required are the position of the first hole, the Z axis movement, the pitch diameter or radius, depending on the control system, and the number of holes required. The computer makes all the necessary calculations to convert the polar coordinates to linear coordinates and to move the slides accordingly.

A variation of this cycle will cater for just two or three holes positioned in an angular relationship to one another. An example is detailed in Figure 6.24. Again, the pole position is programmed and the cutter will be at this point when the cycle commences. The additional dimensional data that have to be supplied are the Z axis movement, the polar radius and the polar angle(s), and the number of holes required, the computer then converts this information to slide movement in the appropriate axes.

On some control systems it is possible to “rotate” more complex loop programmed features such as the example shown in Figure 6.25.

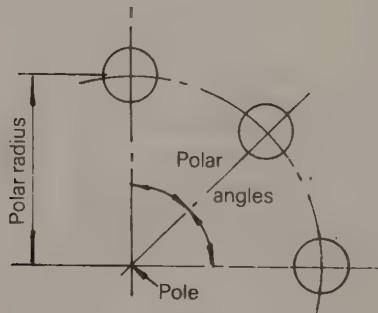


Figure 6.24 Polar coordinates.

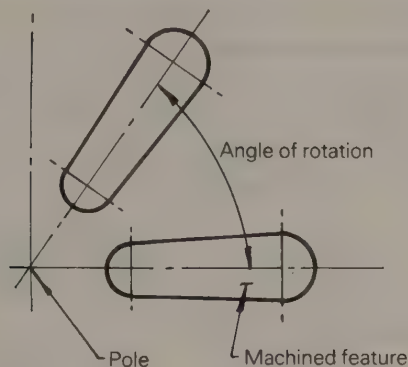


Figure 6.25 *Feature rotation.*

Cycles Devised by the Part Programmer

Cycles devised by the part programmer may be defined as follows. First, there are cycles that are devised specifically for one particular machining task. Second, there are those that may be used when machining a range of components.

Consider the component shown in Figure 6.26, which has a repetitive feature, namely, the recess. When writing a program for machining this particular component, the programmer would devise a cycle, in situations such as this being referred to as a “routine,” for producing just one recess. Via an appropriate call the blocks of data defining the routine can be activated as and when required within the main machining program at new locations.

The construction of a routine may include subroutines also specifically constructed by the part programmer and may also utilize any fixed or canned cycles that are considered appropriate. The technique of programming cycles or routines within routines is referred to as “nesting” and is further described subsequently.

Assume the component shown in Figure 6.26 is quite large so that within each recess there were also a number of holes arranged in three groups, as

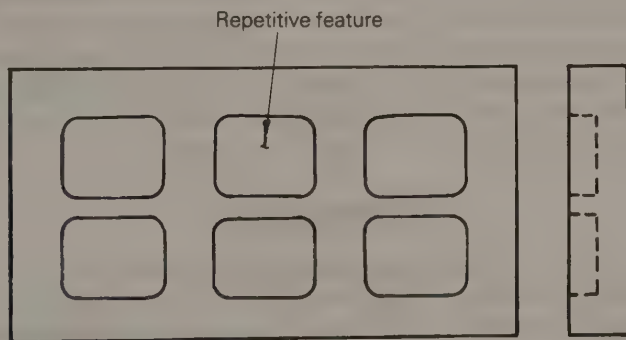


Figure 6.26 *Component with repetitive feature.*

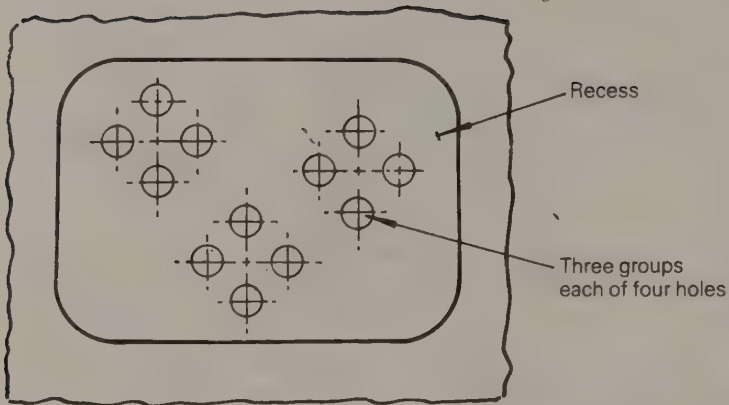


Figure 6.27 *Enlarged detail of component in Figure 6.26.*

shown in 6.27. The main routine would be the data necessary for the production of the recess, as explained above. The subroutine would be the data necessary to produce a group of four holes. The subroutine would be nested within the main routine and called into the main program on three occasions.

However, the production of the four holes is repetitive, and thus it is possible to program to produce just one hole, but to repeat the sequence four times. The complete sequence for producing the component is illustrated diagrammatically in Figure 6.28. On some control systems it is possible to program cycles within cycles as many as eight deep.

Programmer-devised cycles of the second type, to which reference was made above, are useful when a machined feature commonly occurs within the production schedule of a particular company, that is, a machined feature (possibly of unusual design) is required over a range of components. To accommodate this situation some control systems permit routines that are “user defined” to be prepared and “stored within the control system,” so that they may be recalled and utilized as and when required as part of a more comprehensive machining program. A routine of this nature is also referred to as a “macro.”

A macro may have fixed dimensions or it may have parametric variables, that is, the dimensions may be varied to produce different versions of the same basic feature or component. This technique is referred to as “parametric programming” and is described in more detail in Chapter 9.

MIRROR IMAGE

A commonly occurring aspect of mechanical engineering design is the need for components, or features of components, that are dimensionally identical but geometrically opposite either in two axes or in one axis. By using the mirror-image facility such components or features can be machined from just

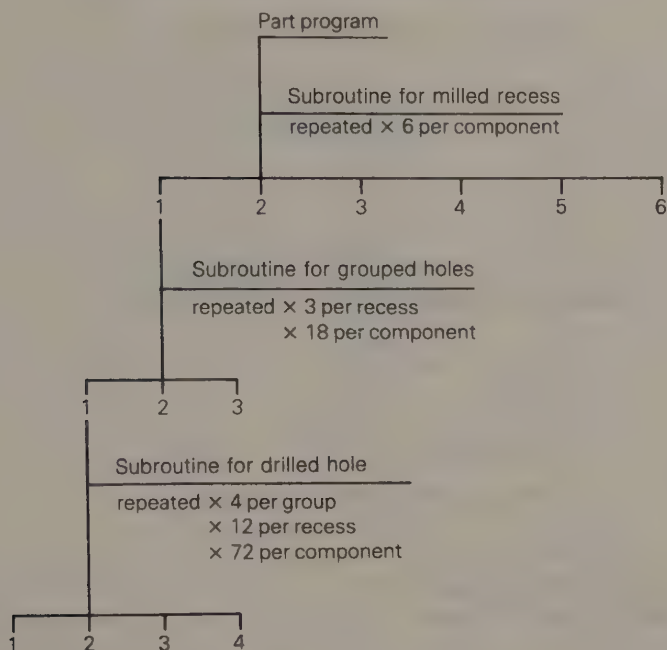


Figure 6.28 'Nesting' three deep.

one set of data. The component shown in Figure 6.29 has a feature that is mirrored in two axes. Note that, to produce the second profile, the positive incremental values become negative and the negative incremental values become positive. To produce a feature of the opposite hand, as shown in Figure 6.30, the direction of slide movement changes in one axis only.

SCALING

Another common requirement in mechanical engineering design is components with the same geometrical shape but varying dimensionally. Figure 6.31 illustrates two such components. When a control system is fitted with a scaling facility, it is possible to produce a range of components, varying in size, from one set of program data. The facility can also be used to produce geometrically identical features of components that may be required to be reproduced to different sizes.

SLASH DELETE

The slash or block delete facility enables part, or parts, of a program to be omitted. It is particularly useful when producing components that have slight

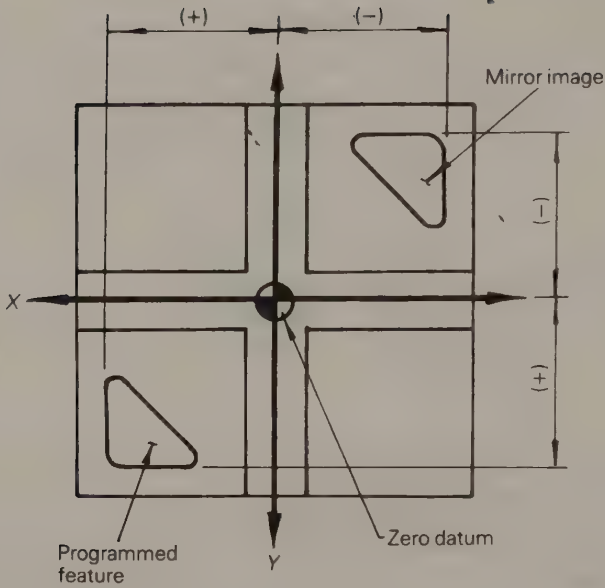


Figure 6.29 Mirror image in two axes.

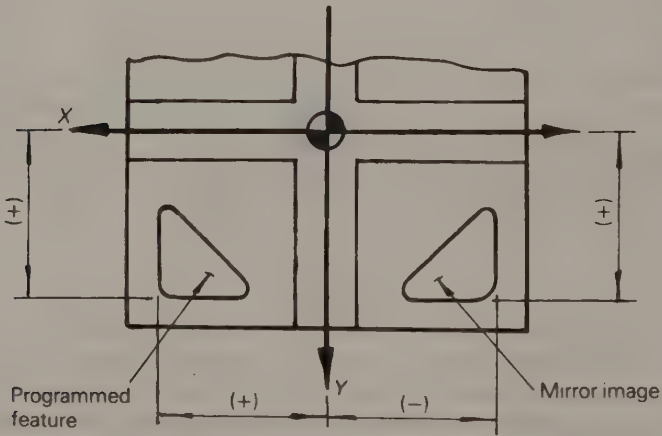


Figure 6.30 Mirror image in one axis.

dimensional variations. For example, a hole may be required in one version of a component but not in another, although all other details may be identical. The program data relating to the production of the hole are contained within the programmed symbols/, one at the start of each block concerned. An example is shown below. See manufacturers' programming manual for possible variations in format.

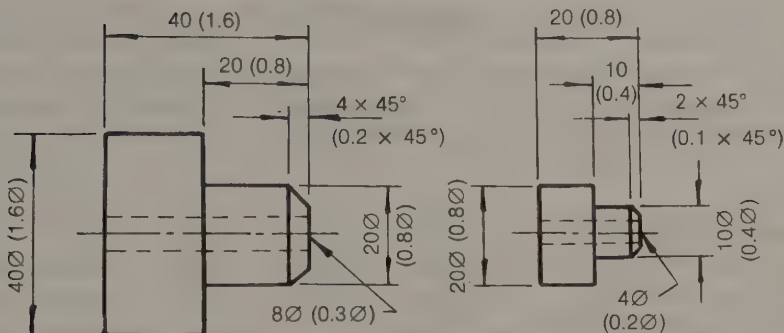


Figure 6.31 Geometrically identical components suitable for production by scaling. (Inch units are given in parentheses.)

```

/N05 G01 Z1000 F 150*
/N06 G00 Z-1000 *

```

To make a component *with* the hole, the operator need not take any action. To produce a component *without* the hole, the operator will have to activate the slash delete switch on the control console at the start of the program. When the slash is reached, the control unit will ignore the data that follow. On some systems, if the slash delete is not activated, the program will stop when the slash is reached and the operator then has to make a positive response either to activate the data or to delete them.

This facility is particularly useful when machining castings or forgings, where stock removal requirements may vary, the operator being given the option to include an extra cut or delete it as necessary.

JOG

The jog facility enables the machine operator to move the machine slides manually via the control console. This may need to be done for a variety of reasons, the most obvious one being when establishing datums at the initial setting of the machine. There are also two standard “G” code boring cycles that call for jog retracts after the cycle before returning to automatic operation. It may also be necessary to stop an automatic sequence and move the machine slides to facilitate work measurement, tool changing due to breakage, and so on. Whatever the reason it is desirable that the automatic program is restarted at the point at which it was interrupted, and most control systems have a *return from jog* facility that returns the machine slides to their original positions, this facility being activated manually via a button on the control console.

PROGRAM STOP

Stops in a machining sequence can be predetermined and included in the part program as a miscellaneous function (M00). Scheduled stops for measurement, tool changing (on manual tool change machines), etc., have to be notified to the machine operator so that he or she will be aware of his or her duties at this point.

Program stops can also be optional, that is, the sequence does not have to stop. Optional stops are also included in the part program as a miscellaneous function (M01) and the control will ignore the command unless the operator has previously activated a switch on the control console.

DATUMS

Machine Datum

The machine datum, also referred to as "zero datum" or simply as "zero," is a set position for the machine slides, having a numerical identity within the control system of zero. All slide movements are made in dimensional relationship to this datum as indicated earlier in the chapter, when absolute and incremental positioning moves were discussed.

On some machines the zero datum may be a permanent position that cannot be altered. On other machines a new zero is readily established by moving the slides so that the cutting tool is placed in the desired position in relation to the workpiece and then pressing the appropriate zero button on the control console. The facility to establish a datum in this manner is referred to as a floating zero or zero shift. The location of the original zero is not retained within the control memory.

A fixed machine datum may be helpful to the part programmer, especially when the programming is carried out remote from the machining facility, since the position can be taken into consideration when writing the program. It will be necessary, however, for the programmer to specify the exact location of the component in relation to the machine datum if the program is to achieve the desired results.

A floating zero affords greater flexibility when machine setting, since the work can be positioned anywhere within the range of slide movement and the zero established to suit. But this can be time-consuming and, if incorrectly carried out, may result in machining errors.

Program Datum

The program datum or zero is established by the part programmer when writing the part program, and the program will require all slide movements to be made in relation to that point.

In practice, the machine zero and the program zero are often synchronized by either accurately positioning the work or, when possible, resetting the machine zero. Any unavoidable variations between the two positions can be accommodated by using the zero offset facility, if available, as described below.

ZERO OFFSET

The zero offset facility enables a machine zero datum to be readily repositioned on a temporary basis. Once it has been repositioned, the slide movements that follow will be made in dimensional relationship to the new datum. It is particularly useful when the original machine datum does not coincide with the part program datum, a situation that can arise, for instance, when a part program has been prepared without regard to the normally fixed position of the machine datum and difficulties are encountered in positioning the workpiece to suit the part program. A simple example of this is when a part program for a turned component has been prepared using the forward face of the workpiece as a zero datum when the machine zero is, as is often the case, located at the back face of the chuck. Use of the zero offset facility will reestablish the zero at the work face. The zero offset facility also enables two or more components to be machined at one setting from the same part program. In Figure 6.32 component 1 would be machined with slide movements made in relation to datum 1. On completion of the machining sequence, the machine table would be caused to move, via the part program or by manual intervention at the keyboard to datum 2 where the offset feature would be activated to the predeter-

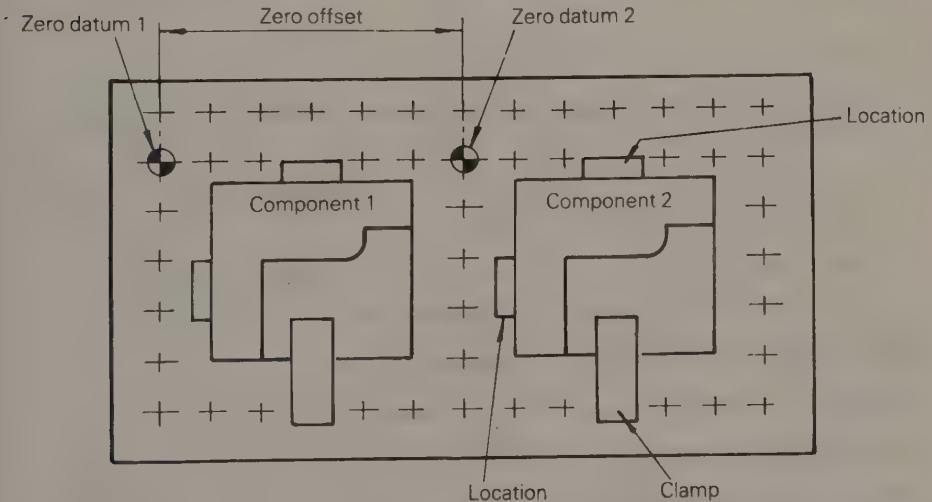


Figure 6.32 The use of zero offset facility when machining components located on a grid plate.

mined offset dimension and the machining sequence would be repeated for component 2, with all slide movements being made in relation to datum 2. The control would retain information regarding the location of the original datum, which remains a permanent feature of the part program, and the machine slide can be caused to return to that position.

The facility may also be used to machine identical features on components of different lengths. A simple example would be to cut a screw thread of particular dimensions on the ends of two bars the overall lengths of which are not the same. Similarly, the facility can also be used when turned workpieces are reversed for secondary operations and the second setting in relation to the Z axis zero differs from the first.

On the more sophisticated controls it is possible to establish a new zero, or zeros, at various stages throughout the program. All subsequent moves will be made in relation to the new datum, but these moves are not necessarily a repeat of the moves made before the new zero was established. This facility enables the features of complex or very long components to be machined by relating slide movement to more than one datum, thus simplifying programming and possibly reducing machining time by limiting the length of slide travel.

QUESTIONS

- 1 What is a preparatory function and how is it designated in word address programming?
- 2 What is a miscellaneous function and how is it designated in word address programming?
- 3 What is meant by the term "modal"?
- 4 Name and describe the three types of positioning control used on computer numerically controlled machine tools.
- 5 Explain, with the aid of a simple sketch, the difference between absolute and incremental dimension definition.
- 6 When are the letters *I*, *J*, and *K* used in a word address program?
- 7 What is meant by the term "ramp suppression"?
- 8 Describe what happens during a peck drill cycle.
- 9 What is a looping cycle and when is it used?
- 10 Describe what happens during a pocket milling cycle.
- 11 What is a bolt hole circle?
- 12 What is meant by the "rotation" of a machined feature?

- 13** With the aid of a simple sketch describe the effect of reproducing a machine feature using the mirror image programming facility in (a) two axes; (b) one axis.
- 14** What is the programming function that permits the production from one set of data, components geometrically identical but with proportional dimensional variations?
- 15** Explain the meaning of the term “nesting” as applied to machining cycles.
- 16** When is the block or slash delete facility likely to be used and how is it generally invoked?
- 17** What is the jog facility on a machine control system and when is it likely to be used?
- 18** Give two reasons for including an optional stop in a program.
- 19** Why is it necessary to inform a machine operator of the scheduled stops in a machining program?
- 20** With the aid of simple sketches describe the meaning of zero offset.

SPEEDS AND FEEDS FOR NUMERICALLY CONTROLLED MACHINING

CUTTING SPEEDS AND FEEDS

It is difficult to determine precise data for any metal-cutting operation without knowledge of the practicalities involved. For example, the condition of the machine, the power available, the rigidity of tooling and work-holding arrangements, the volume of metal to be removed, the surface finish required, and the type of coolant to be used are all factors that have to be considered when determining the appropriate speeds and feeds to be used. The programmer is obliged to make program entries that will be, if not perfect, then at least functional.

Should it prove that the programmed speed and feed are inappropriate, machine control units have manually operated override facilities to enable the operator to increase or decrease speeds and feeds as machining proceeds. If such action is found to be necessary, the programmer should be informed so that the part program can be modified accordingly.

The selection of appropriate speeds and feeds can be based on experience when the programmer possesses the necessary practical background. Experienced craftsmen often have an instinctive ability to recognize the correct speed and feed for any particular machining task, having long forgotten any theoretical basis they may have once been taught concerning such matters. But this approach is perhaps more suitable to conventional machining processes than it is to CNC machining.

Reference to data published by the manufacturers of the cutting tools to be used is more appropriate for numerically controlled machining, since the lack of operator involvement that is a feature of machining in this way makes it imperative that the optimum speeds and feeds are used. However, when reference is made to published data, the programmer should exercise caution. The figures quoted, while perfectly feasible when used under the correct conditions, can appear to be somewhat optimistic when applied to many machining situations. Examples of manufacturers' data relating to cutting speeds and feeds are included in Appendices B.

Whatever approach is used, it is essential that the programmer fully appre-

ciates the capabilities of any machine for which he or she is preparing programs. For instance, it is pointless to program speeds and feeds that result in metal removal rates that are beyond the power capacity of the machine. Conversely, it is equally pointless to underuse the power available. If a programmer lacks essential knowledge of this nature, then liaison with those who have had practical experience of the machine should be a priority.

Unusual machining situations involving set-ups that may lack rigidity should be approached with care, and the programmed speeds and feeds should be initially on the low side. They can always be increased later in the light of experience gained at the machine when metal cutting has actually taken place.

Special attention should also be given to specific requirements regarding surface finish. Machinists are often obsessed with obtaining a "good" finish, but it should be remembered that there is no point in reducing metal removal rates to obtain a high-quality surface finish that is not necessary. A further point to remember is that a designer may specifically require a "rough" finish, so it is essential to work to the information contained on the drawing. The programmer should ensure that the setup person/operator responsible for the machining operation fully appreciates what the requirements are.

The intricate contouring capabilities of CNC machines can also present problems regarding surface finish. For example, feed rate that produces satisfactory results when machining a parallel turned surface may no longer produce acceptable results when the cutting tool changes direction to machine a tapered or radial surface. Similar situations present themselves during milling operations. When machining complex profiles that are subject to stringent surface finish requirements, the results obtained from programmed feeds and speeds, particularly feeds, should be monitored and modifications made if necessary.

Surface finish is, of course, affected by the condition of the cutting tool. Tool performance begins to deteriorate from the moment the tool is first used, and ultimately it not only affects the surface finish but may result in dimensional features not being maintained, unacceptable vibration, work deflection and eventually total failure of the tool. An unsuitable choice of cutting speeds and feeds may hasten this process, so the part programmer must give due consideration to tool life when making decisions in this respect.

It is possible to calculate the life of a cutting tool. The formula used for this purpose was derived by experimentation and, because of the many variables that exist between one machining situation and another, the results obtained from its application can only be used as a guide. The reader may reasonably ask why bother to make such a calculation if the result is only a guide, and it would be difficult to give a totally convincing answer. Nevertheless, it would be helpful to know when a tool was reaching the end of its life, so that replacement could be effected before total failure occurred. The automatic tool condition sensing devices being applied to the more complex CNC machining installations may provide the answer. In the meantime, how is the part programmer to decide what speeds and feeds to use to give an acceptable tool life,

while at the same time achieving the basic objective of removing metal in the least possible time?

As stated earlier, reference can be made to the cutting tool manufacturers' literature. Their figures will have taken into account the fact that a reasonable tool life is required by the user of their products. But there is still the problem that local conditions may make their recommendations invalid. Yet again, it may be initially necessary to rely on past practical experience, and to be prepared to make modifications based on a reasoned appraisal of the situation when metal cutting is under way.

One further point relating to tool performance should be noted. The chip-breaking qualities of most carbide-tipped tools is directly related to speeds and feeds. If chip clearance becomes a problem, some modification of the cutting conditions may be appropriate.

SPINDLE SPEEDS

The program data controlling spindle speeds is expressed in one of two ways, the numerical value in both cases being preceded by the letter S. Thus a data entry of, say, S250 could indicate either a *constant surface cutting speed* of 250 m per minute or, alternatively, a *spindle speed* of 250 revolutions per minute (rev/min).

All machines have the facility to program a set spindle speed in revolutions per minute, while the alternative facility of programming a constant surface cutting speed is now commonly available on the majority of turning centers.

When both facilities are available, the machine controller differentiates between the two possibilities via a previous data entry that will establish the desired operating mode. In the case of word address programming this mode is established by a G code entry, commonly G96 (ft, or m per min) and G97 (rev/min). Feet or meters per minute will depend on whether the G70 inch programming or G71 metric programming code is active. In the case of conversational MDI, the mode is established by selection from the displayed options.

Consider the process of programming a constant surface cutting speed. The cutting speed is the rate at which the cutting tool passes over the workpiece material, or alternatively the rate at which the material is traveling as it passes the cutting tool. As indicated previously, it can be expressed in either meters or feet per minute.

Appropriate surface cutting speeds for use with cutting tools made of specific materials, when used to cut certain metals, have been determined by experiment. The figures give due regard to the maximum metal removal rates that can be obtained, while at the same time equating satisfactorily with other factors such as tool life, surface finish, and power consumption. These recommended cutting speeds are published by the manufacturers of cutting tools as

a guide to users of their products (see Appendix B). As stated earlier, it may be necessary to modify these values to suit local conditions before making a program entry.

The advantage of programming a surface cutting speed as opposed to a set spindle speed in revolutions per minute is best appreciated by considering the simple operation of parting-off from bar stock during a turning operation.

During a parting-off operation the diameter of the work where metal cutting is actually taking place is steadily decreasing, and therefore the cutting efficiency is only maintained if the spindle speed increases at a corresponding rate. This steady increase, which maintains the most efficient metal cutting rate for that particular job material, is automatically achieved via the constant surface cutting speed programming facility.

The process of parting off is a convenient one to explain the value of the constant surface cutting speed programming. However, it is not a process where the use of such a facility is absolutely critical. The facility is more likely to be of value during the turning of complex profiles requiring a uniformly high standard of surface finish throughout the turned length.

In order to program a constant spindle speed in revolutions per minute, it is necessary to make a simple calculation that takes into consideration the recommended surface cutting speed referred to previously, and also the diameter of the workpiece in the case of turning operations, or the cutter in the case of milling operations. The relationship between these factors is expressed as follows.

$$\text{Spindle rev/min} = \frac{1000 \times \text{Cutting speed in m/min}}{\pi \times \text{Work or cutter diameter in mm}}$$

Multiplying the cutting speed by 1000 converts it from *meters* per minute to *millimeters* per minute, while multiplying the work or cutter diameter by π , that is, calculating the circumference, determines the relative linear travel per revolution in millimeters. Dividing the circumference in millimeters into the cutting speed in millimeters per minute determines the number of revolutions per minute required.

When inch programming, the relationship between spindle speed, work or cutter diameter and the required cutting speed is expressed as follows:

$$\text{Spindle speed in rev/min} = \frac{12 \times \text{Cutting speed in ft/min}}{\pi \times \text{Work or cutter diameter in inches}}$$

In this case multiplying the cutting speed by 12 and so converting it to in./min makes all the units in the equation compatible.

A number of cutting tool manufacturers distribute simple calculators, the use of which eliminates the need to make calculations. These devices will indicate the appropriate cutting tool material, selected from the manufacturer's range, that should be used when machining a certain material type, together with a

recommended surface cutting speed for the type of operation (roughing or finishing) that is to be undertaken.

SURFACE CUTTING SPEEDS

The speeds in Table 7.1 are suitable for average metal-cutting conditions. In practice it may be possible to increase these speeds considerably for light finishing cuts or, conversely, to reduce them for roughing cuts. For more accurate speed and feed tables related to operation, material, and finish refer to *Machining Data Handbook* by the Machinability Data Center (Metcut Research Associates Inc.) or *Machinery's Handbook* by Oberg, Jones, and Horton (Industrial Press Inc.).

SPINDLE SPEEDS

Example: Determine the spindle speed required to turn a 2.5-in.-diameter piece of aluminum using a high-speed tool and a surface feet per minute (SFPM) cutting speed of 300 ft/min:

Spindle speed, rev/min = $\frac{12 \times 300}{\pi \times 2.5} = 458$

Example: Determine the spindle speed required to turn 50 mm diameter brass using a cemented carbide tool and a surface cutting speed of 180 m/min:

Spindle speed, rev/min = $\frac{1000 \times 180}{\pi \times 50} = 1146$

These spindle speeds will be correct for turning a 2.5 in. or 50 mm diameter bar. However, as the diameter decreases, as for instance during an end-facing operation, these spindle speeds are no longer valid or efficient. On many numerical control systems it is possible to program a surface cutting speed only, and the machine spindle speed will automatically vary within the range of the machine to compensate for changes in the work diameter, thus providing a constant cutting speed.

Table 7.1 Surface cutting speeds, m/min (ft/min)

Tool material	Part material			
	Mild steel	Cast iron	Aluminum alloy	Brass
Cemented carbide	170 (300)	100 (225)	250 (900)	180 (450)
High-speed steel	28 (90)	18 (75)	120 (300)	75 (150)

FEED RATES

The manufacturer's calculators referred to previously will also indicate an appropriate feed rate for the operation. The feed rate is the speed at which the cutter penetrates into the work material.

When programming data relating to feed rate, it can be expressed either as millimeters per minute (mm/min) or millimeters per revolution (mm/rev) of the machine spindle. With inch programming, the units will be inches per minute (in./min) or inches per revolution (in./rev).

The letter F is commonly used to denote the feed rate in a part program. Thus F25 could indicate a feed rate of 0.25 in. or mm/rev, while F80 could indicate a feed rate of 80 in. or mm/min, depending on the mode of expression being used. It is necessary to ensure that the data entered and the programming mode are compatible.

Although variations exist, the set-up data using the G codes common to a number or word address programming systems are as follows:

G94 feed/min

G95 feed/rev

In this situation the units to be used are established by the use of G70 for inch and G71 for metric.

Feed rates are published by cutting tool manufacturers in the same way as surface cutting speeds. Usually the rates are expressed as mm/rev or in./rev. To convert to mm/min or in./min involves making a simple calculation as follows:

$$\text{Feed mm/min} = \text{Feed mm/rev} \times \text{Spindle speed rev/min}$$

or,

$$\text{Feed in./min} = \text{Feed in./rev} \times \text{Spindle speed rev/min}$$

The manufacturers of milling cutters sometimes quote recommended feed rates in millimeters or inches per tooth, in which case it is necessary, prior to making the preceding calculation, to determine the feed per revolution of the cutter. This is achieved as follows:

$$\text{Feed/rev} = \text{Feed/tooth} \times \text{Number of cutter teeth}$$

Feed Rates for Turning

Cemented carbide tools are used extensively for turning operations. It is common practice for the manufacturers to quote recommended feed rates in inches or millimeters per spindle revolution (in./rev. or mm/rev). Typical feed rates for different work materials are given in Table 7.2.

To determine the feed rate in mm/min (in./min):

$$\text{Feed mm/min (in./min)} = \text{Feed mm/rev (in./rev)} \times \text{Spindle speed (rev/min)}$$

Table 7.2 Typical feed rates for turning, mm/rev (in./rev)

Mild Steel	Cast Iron	Aluminum Alloy	Brass
0.25 (roughing 0.007–0.025) (finishing 0.002–0.007)	0.25 (roughing 0.007–0.025) (finishing 0.002–0.007)	0.3 (roughing 0.007–0.030) (finishing 0.002–0.007)	0.3 (roughing 0.007–0.030) (finishing 0.002–0.007)

When a constant surface cutting speed is programmed, that is, the spindle speed varies automatically to compensate for variations in the work diameter, the feed rate is programmed in in. or mm/rev to maintain a constant feed rate per spindle revolution. When the spindle speed is programmed at a constant rev/min, the feed rate can be entered either as in. or mm/rev or in. or mm/min, since both will result in a constant relationship between surface cutting speed and feed.

Feed Rates for Milling

The manufacturers of milling cutters state recommended feed rates as in. or mm/rev, in. or mm/min or in. or mm/tooth.

When feeds are quoted as in. or mm/rev or in. or mm/min, they usually refer to specific cutters in the manufacturer's range and cannot be generally applied. For instance, if two face mills both of the same diameter, but one having five carbide inserts and the other six, were used at the same spindle speed with a feed quoted per revolution, it would mean that the cutter with the fewest teeth would be subjected to a much higher volume of metal removal per tooth than the cutter with more teeth. So for general use feed rates quoted in in. or mm/tooth are more suitable. These data can then be used to determine the feed rate per revolution as follows:

$$\text{Feed, mm/rev (in./rev)} = \text{Feed, mm/tooth (in./tooth)} \times \text{Number of teeth}$$

And from this formula,

$$\text{Feed, mm/min (in./min)} = \text{Feed, mm/rev (in./rev)} \times \text{Spindle speed (rev/min)}$$

Typical feed rates are given in Table 7.3.

Feed Rates for Drilling

High-speed-steel drills are used extensively for producing smaller holes. Since small-diameter drills are liable to break, the feed rate is related to the drill size. Typical feed rates are given in Table 7.4.

Cemented carbide drills, sometimes with tips brazed to a medium carbon steel shank, but more commonly as clamped inserts, are favored for larger

Table 7.3 Typical feed rates for milling, mm/tooth (in./tooth).

Work Material	High-speed-steel cutters		Cemented carbide cutters	
	Face and Shell End Mills	End Mills	Face and Shell End Mills	End Mills ^a
Mild steel	0.25 (0.010)	0.15 (0.005)	0.30 (0.008–0.020)	0.18 (0.003–0.010)
Cast iron	0.30 (0.013)	0.18 (0.007)	0.50 (0.008–0.020)	0.21 (0.005–0.012)
Aluminum alloy	0.40 (0.022)	0.17 (0.011)	0.60 (0.005–0.020)	0.25 (0.005–0.020)
Brass	0.35 (0.014)	0.15 (0.007)	0.40 (0.005–0.020)	0.19 (0.003–0.012)

^a As with high-speed-steel end mills, the design of some cemented carbide end mills provides for both plunge and side cutting.

Table 7.4 Typical feed rates for high-speed steel drills

Drill Size (mm)	2	4	6	8	10	12	14	16	18	20
Feed Rate (mm/rev)	0.05	0.10	0.12	0.15	0.18	0.21	0.24	0.26	0.28	0.30
Drill size (in.)	0.125	0.250	0.500	1.0	1.0					
Feed rate (in./rev)	0.001–0.003	0.002–0.006	0.004–0.010	0.007–0.015	0.010–0.025					

holes. The feed rates for these drills compare with those used for carbide insert end mills. Carbide use in small holes requires a solid carbide tool, which is very expensive.

The feed rate for solid carbide drills can be determined by using tables for brazed tip as a starting point.

As with turning and milling, the feed in in. or mm/rev can be used to determine the feed in in. or mm/min, as follows:

$$\text{Feed (in. or mm/min)} = \text{Feed (in. or mm/rev)} \times \text{Spindle speed (rev/min)}$$

FEED RATE AND SPINDLE SPEED OVERRIDE

On the majority of control systems there are facilities that enable the operator manually to change, via a setting dial, both the feed rate and the spindle speed. Selection is usually on a percentage basis, 0–150% being fairly typical.

Changes in speed and feed may be necessary for a number of reasons. For example, the operator may judge that the rate of metal removal can be safely

increased and so may increase the feed rate. Similarly, the operator may judge that a prolonged tool life would result if the spindle speed were decreased, or the surface finish being obtained from the programmed feed rate may be unsatisfactory, and so on. Manual control of speeds and feeds is also a very helpful feature during machine setting and program proving.

Changes made manually do not affect the basic program, although if an operator decides that the programmed feed rate or spindle speed for any part of a program is unsatisfactory he or she should make the fact known so that a permanent change can be made.

QUESTIONS

- 1 Select a suitable surface cutting speed and calculate the spindle speed required to turn a mild steel component with a diameter of 55 mm, using a cemented carbide indexable insert cutting tool.
- 2 If the component in question 1 has a second diameter of 24 mm, what change in spindle speed will be necessary?
- 3 Compare the surface speeds for machining mild steel using high-speed steel and cemented carbide cutters and express the variation as a percentage.
- 4 What is meant by the term "constant surface cutting speed"? Quote a situation where it may be desirable during a machining operation.
- 5 Select a suitable feed rate in in./rev for finish turning free-cutting mild steel using a cemented carbide tool. Using these data, determine a suitable feed rate in in./min for turning a diameter of 3.5 in.
- 6 Select a suitable feed rate in mm/rev for finish turning cast iron using a cemented carbide tool. Using information found in previous tables, determine a suitable feed rate in mm/min if turning a diameter of 75 mm in the same cast iron material.
- 7 Select a suitable surface cutting speed and determine the spindle speed required to drill a 9/16 in. diameter hole in mild steel.
- 8 Select a suitable surface cutting speed and determine the spindle speed required to drill a 6 mm diameter hole in free-cutting brass.
- 9 Why is it that the feed rate in in. or mm/rev for drilling operations varies with the drill diameter?
- 10 Calculate the approximate feed rate in mm/min for milling a slot in brass with a high-speed steel cutter of 15 mm diameter.
- 11 Calculate the approximate feed rate in in./min for milling a slot in an aluminum part with a high-speed steel two flute cutter of 0.375 in. diameter.

- 12 Determine a suitable spindle speed for face milling aluminum alloy using a cemented carbide cartridge-type milling cutter of 100 mm diameter.
- 13 Determine a suitable spindle speed for face milling cast iron using a cemented carbide insert type milling cutter with a 4.5 in. diameter.
- 14 If the cutter in question had six cartridges, what would a suitable feed rate be in mm/rev?
- 15 If the cutter in question 13 had eight inserts, what would a suitable feed rate be in in./rev?

8

PART PROGRAMMING FOR COMPUTER NUMERICALLY CONTROLLED MACHINING

THE PART PROGRAM

The term part program is used to describe a set of instructions that, when entered into a machine control unit, will cause the machine to function in the manner necessary to produce a particular component or part. Manual part programming is the term used to describe the preparation of a part program without recourse to computing facilities to determine cutter paths, profile intersecting points, speeds and feeds, etc.

The program may be prepared manually and expressed in a coded language that is applicable to the machine controller being used. Alternatively, it may be written in another language or compiled by the use of computer graphics. The result is then post-processed, or translated, to suit the machine controller.

Included in the part program will be the necessary dimensional data relating to the features of the component itself, together with control data that will result in the machine making the slide movements required to produce the component. These data will be supplemented by instruction data that will activate and control the appropriate supporting functions.

Programs as entered into machine control units involve either of two programming concepts:

- (a) word address
- (b) conversational manual data input (MDI)

There are considerable variations between the two methods.

Whether a production scene incorporates total automation or merely one or two numerically controlled machines positioned among traditional machines, at the heart of successful numerical control is efficient competent part programming. The practical *skill* level requirement on the shop floor is, without doubt, in decline, but a high level of practical *knowledge* is essential if part programmers are to use costly equipment at their disposal to the best advantage. The selection of a correct sequence of operations, together with efficient cutting speeds and feeds, tooling and work holding, and the ability to express these requirements in the correct format are of paramount importance.

Unfortunately, programming methods differ and even when the basic approach is similar (for example, with word address), there are still variations and peculiarities, and conversational manual data input is very individual.

Thus the reader should appreciate that the ability to program with one control system, although there is much carryover, rarely means that knowledge can be used *in total* elsewhere. Specialist training is essential, and most machine-tool manufacturers respond to this by offering training courses as part of the overall package to customers buying their equipment. However, once the basic concepts involved in part programming are understood, the change from one system to another does not appear to be a major problem. Indeed, the variations encountered can be a source of much interest, while the mastery of yet another system can give considerable personal satisfaction.

PROCEDURE

Taking as a starting point the detail drawing of the component to be manufactured, the tasks that confront the part programmer may be listed as follows:

1. Select a machine capable of handling the required work.
2. Determine the machine process to be used.
3. Determine work holding and location techniques.
4. Determine tooling requirements and their identity.
5. Document, or otherwise record, instructions relating to work holding, work location, and tooling.
6. Calculate suitable cutting speeds and feed rates.
7. Calculate profile intersecting points, arc centers, etc.
8. Determine appropriate tool paths including the use of canned cycles and subroutines.
9. Prepare the part program.
10. Prove the part program and edit as necessary.
11. Record the part program for future use.

Although these stages have been given a separate identity, they are very much interrelated and cannot be treated in isolation. A diagrammatic impression of the approach to be adopted is given in Figure 8.1.

MACHINE SELECTION

In selecting the machine to be used the first consideration is the type of work that has to be carried out. The tolerance and surface finishes required on the part will determine the type of machine and process to be used. Even when the type of machine is established, its specifications will need to be reviewed to ensure that part accuracy can be maintained.

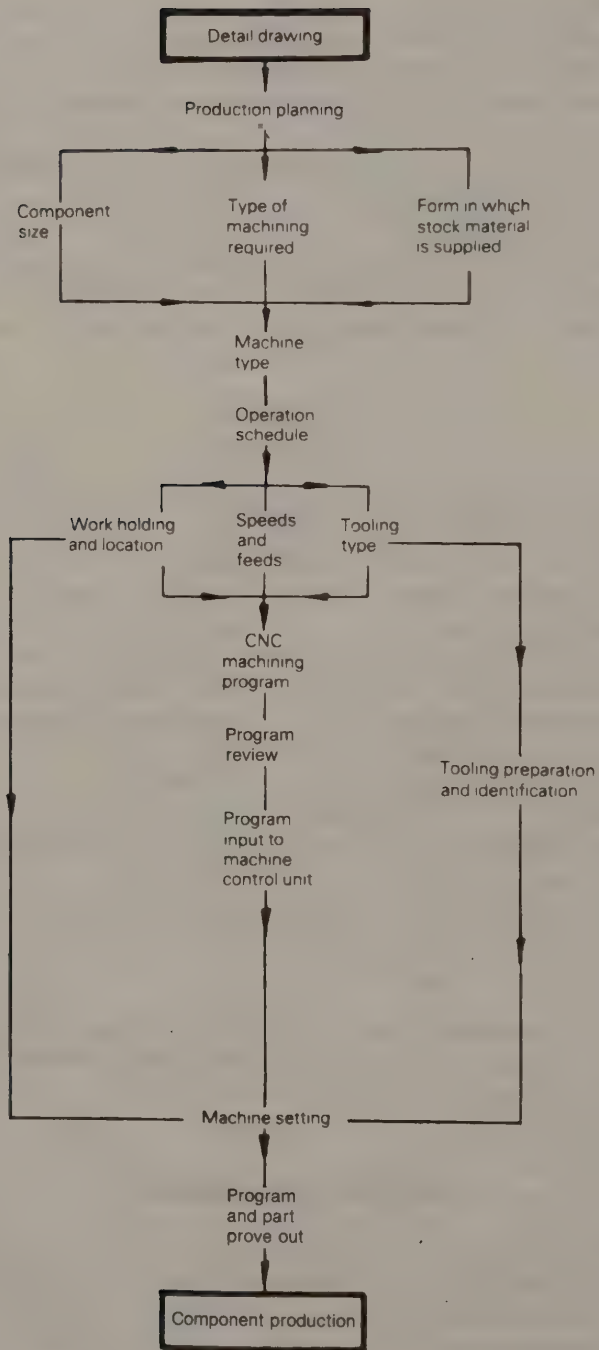


Figure 8.1 Procedures associated with part programming.

For relatively simple components the choice will be obvious and is likely to involve just one machine. On the other hand, more complex designs may require machining to be carried out on a second or perhaps third machine. It may be necessary to move from one machine to a second before returning to the original machine for further work, and so on.

Such transfers, and the stage at which they will take place, need to be determined clearly, since they will have a direct bearing on the preparation of appropriate machining programs.

Machine selection will also be influenced by component size, and the programmer must ensure that any machine used has the necessary physical capacity to accommodate the workpiece.

Decisions made at the component design stage relating to materials—whether the form of supply will be a casting or a solid bar, for instance—will also need to be considered, since this may have some bearing on work-holding and machine-loading arrangements.

PROCESSING OF MACHINING OPERATIONS

Having selected a machine capable of handling the required work, the next task confronting the part programmer is to decide on a suitable sequence of operations.

In order to do this effectively the programmer should ideally have a thorough understanding of the capabilities and operating procedures associated with the particular machine to be used, and adequate knowledge of the work-holding equipment and tooling that can be employed.

It is often the case that, giving due regard to safety requirements, a machining task can be tackled in more than one way with equally good results in terms of dimensional accuracy and surface finish. But the programmer must always bear in mind one objective is to complete the machining as quickly and efficiently as possible. There are two basic planning techniques that, when carefully considered, can make a significant contribution to achieving this objective.

The first is to carry out as much machining as possible at one work setting and to avoid unnecessary repositioning of the work, since this can be a very time-consuming business. The second is to carry out as much machining as possible with each cutting tool called, and to avoid unnecessary tool changing or indexing. The programmer should bear these points firmly in mind when listing the sequence of operations to be adopted.

The compilation of the process of operations to be used will not only be an aid to logical thinking throughout the rest of the part programming process, but it is also likely to be of value to the machine operator and may be required as a record for future reference. The more complex the component, the more vital the compilation of the process becomes.

It is likely that the operations process will form just part of the general documentation relating to a particular job, which will also contain information relating to work-holding, tooling, speeds, and feeds. The documentation relating to these aspects of part programming are discussed subsequently.

WORK-HOLDING AND LOCATION

The part programmer's responsibilities regarding work-holding and location are as follows:

- (a) determine the work-holding device or devices to be used;
- (b) determine if there will be a need to use supplementary support at any stage during a machining sequence;
- (c) determine the means of ensuring accurate location of the workpiece prior to machining;
- (d) document all matters relating to work setting that will have a direct effect on the validity of the part program and that will, therefore, be of importance to the machine set-up person.

Decisions made in relation to these factors are greatly influenced by component shape and size. Components of regular shape are usually accommodated in standard work-holding devices such as chucks, collets, and vises. Components of irregular shape often require special work-holding arrangements, and as a result demand extra attention from the programmer. He or she may find it necessary to include special slide movements in the program, solely to avoid collisions between the cutting tools and the clamping.

Similarly, the programmer will need to give special attention to components requiring supplementary support—the use of a center support or steady rest, for example—and may well have to include control of these features within the part program.

Multicomponent settings will also have a direct effect on the approach adopted when preparing the part program.

A special characteristic of CNC machining involving very high rates of metal removal is that considerable cutting forces may be exerted in a number of directions during the production of a single component, with very rapid change from one direction to another, possibly occurring without the safeguard of manual observation or intervention. This variation in cutting force direction means that the prime objective in work location, that of ensuring that the cutting forces are directed against an immovable feature in the work-holding arrangement, may not always be met when using standard equipment. For example, work held in a conventional machine vise is only positively located when the cutting forces are directed against the vise jaw. If the cutting force changes direction so that it is at 90° to the fixed jaw, there will be a frictional hold only, which is not foolproof.

When confronted with the problem of multidirectional cutting forces, the programmer should give full consideration to the alternative approaches available. Devices such as the grid plate will provide for positive location in several directions, but it may be necessary to use a specially devised fixture. A number of the project components included in Appendix C will require this approach.

It is possible that the work-holding equipment available is very limited in range, such as a machine vise. In this situation the programmer will have to make the best of arrangements such as the frictional hold described previously. For example, a reduction in metal removal rates will reduce the cutting forces exerted on the workpiece. Each problem encountered will require individual assessment, and the methods used to overcome the problem should be selected with reference to the high safety standards that are so essential in CNC machining.

Another factor that must be considered is that of geometric tolerances, as listed in Appendix D. When any of these are encountered on a part drawing, the programmer must ensure that the work-holding and location arrangements being used will enable them to be achieved. It is a further area of part programming that requires the programmer to be well versed in the practical side of CNC machining, and to have a full understanding of the capabilities and limitations of the work-holding devices that may be used.

In order that specified geometric requirements are satisfied, it may be necessary to adopt a special approach to work setting, or, as is more likely, work resetting before carrying out further operations. In such cases it is imperative that the part programmer indicates to the machine set-up person or operator his or her reasons for doing so. Such information is included in the general documentation relating to that particular workpiece.

The importance of positive location of the workpiece to absorb the forces exerted by the metal-cutting action has already been stressed. There is, however, another reason why the part programmer is concerned about precise location of the work. He or she will program the slide movements in relation to a datum that will be determined when the part program is prepared, and unless the part to be machined is precisely positioned in relation to that datum the intended machining features will not be achieved. Subsequent parts must also be positioned in exactly the same way to ensure uniformity of the product.

When establishing a program zero datum, the programmer will have to take into consideration the reference zero position that is an incorporated feature of the machine control system. The machine zero may or may not be in a fixed position. If it is fixed, it may be capable of being shifted on a temporary basis via the part program using the G92 preset code. It may be capable of being established anywhere within the operating range of the machine, or there may be limitations on repositioning. Whatever the circumstances, the programmer will need to understand them completely.

Consider first a control system that permits a machine zero to be established anywhere the programmer chooses. In this situation it may be considered that

the correct programming approach is to establish a machine zero that will correspond with the chosen program zero. So for a component such as the one illustrated in Figure 8.2 the programmer selects the corner of the workpiece as zero for all programmed moves in the X and Y axes, and a 2 mm (0.1 in.) clearance between the top of the work and the Z axis zero. By selecting the upper left-hand corner of the part as program zero and machine zero, the programmer can use many part print dimensions in the part program. To ensure that there is correlation between the two zero positions the following machine setting approach will be necessary.

1. Set the corner of the vise jaw to zero in the X and Y axes (achieved by using a center locator or wiggler, or possibly an electronic probe).
2. Set the Z axis zero 2 mm (0.1 in.) above the work surface (achieved by touching on to a suitable worksetting block and calibrating the tool length offset accordingly).
3. Locate all workpieces using the corner of the fixed jaw of the vise as a reference position. (A plate attached to the vise jaw may be used to simplify this process).

The setting arrangement that accommodates the X and Y axes requirements is illustrated in Figure 8.3.

Consider now a situation involving a turned component such as that illustrated in Figure 8.4, and assume that the programmer has chosen to establish the face of the part as the Z datum zero and that the machine spindle center line is the X axis zero, as is normal. All that is required of the programmer is to ensure that the machine set-up person or operator is aware that the program datum is at the face of the work. The set-up person or operator will be required

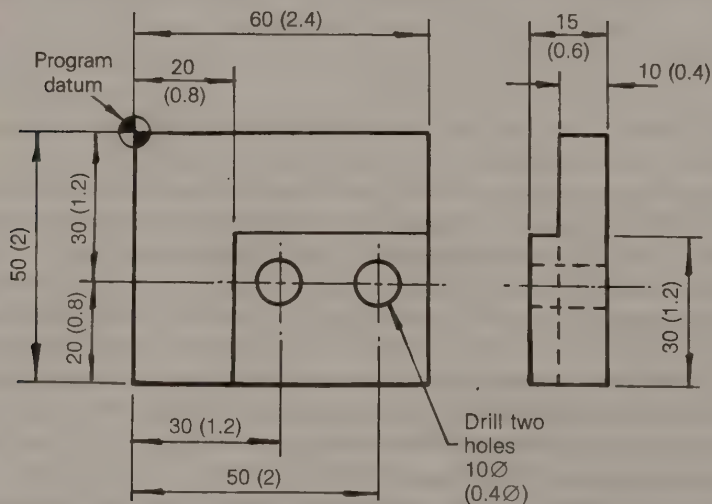


Figure 8.2 Component detail. (Inch units are given in parentheses.)

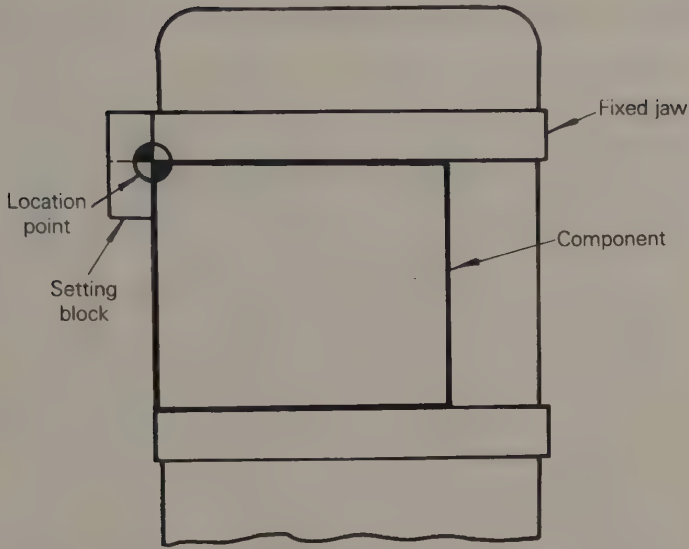
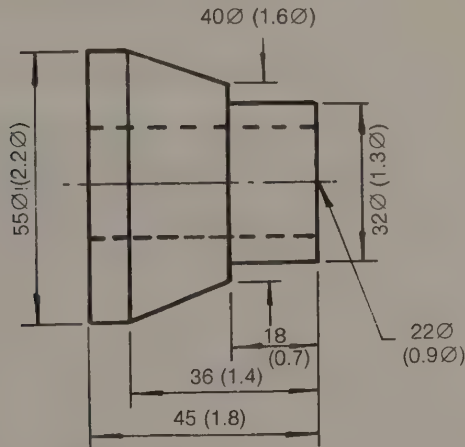


Figure 8.3 Use of stop block on fixed jaw for component location.



Material: medium carbon steel

Figure 8.4 Component detail. (Inch units are given in parentheses.)

to establish the Z axis zero at the machine in the manner appropriate to that particular machine, and then ensure that all workpieces are all set to a measured overhang or to a stop as illustrated in Figure 8.5.

It is often the case that turning centers have a set zero datum for the machine, usually at the back face of the chuck or a reference surface on the spindle nose.

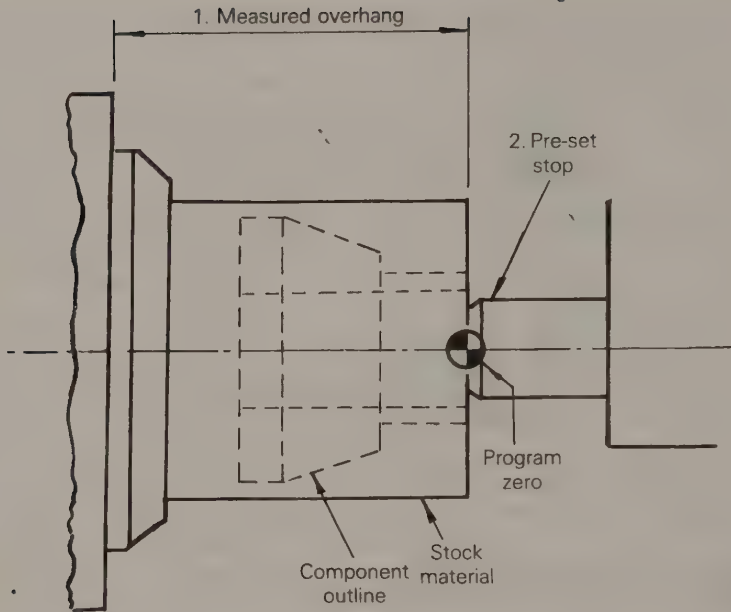


Figure 8.5 Alternative work setting techniques to establish a datum for turned work.

This type of zero cannot be changed but can be shifted on a temporary basis using the G92 preset axis code.

The programmer may choose to use the back face of the component as the program zero in the Z axis, a technique often applied when work is being produced from prepared billets. Work location is simple, and simply involves ensuring the material is firmly placed against the reference face. A further bonus is that all programmed slide movements will be positive.

To use the facility of repositioning the zero on a temporary basis—so that it corresponds to a program zero established at the workpiece face for instance—it will be necessary for the programmer to determine the amount of shift required to accommodate all the programmed movements in the Z axis. The dimensional value of the shift required, that is, the work overhang, must be documented. Eventually it will be entered into the program through the use of the G92 code followed by X, Z, and/or Y axis positional data as to what current slide positions should be. Some older machines may still use another method of establishing zero shifts through special offset tables, which can be activated by assigned codes like E, F, or H.

To ensure that the programmed machine movements achieve the desired effect, the work material has to be positioned accurately, either manually or automatically against stops, and this function is the responsibility of the machine set-up person or operator. The accuracy of this method of work setting can be improved if the overhang is slightly larger than the actual work requires, al-

lowing a facing cut to be used early in the machining sequence to establish the new zero precisely.

It may be necessary to provide for more than one zero shift within the same turning program. A common situation is when the component length is such that, to ensure adequate support and to avoid chatter, part of the machining is carried out with a reduced overhang. After a programmed stop in the machining cycle, the operator repositions the work to suit the second zero position. Alternatively, the repositioning of the work may be achieved automatically through the program. This is particularly appropriate when a bar feed is utilized, the bar feeding to appropriate stops. The provision of a center support may also be a feature of such an arrangement. After the second zero shift all subsequent moves will be made in relation to that datum.

An example of a component which would involve two zero shifts during machining is shown in Figure 8.6. Because the diameter of the component is relatively small in proportion to its length, it would be advisable to use two settings and a center support for the second sequence of machining operations. The first setting involving the shift of the machine zero to the work face is illustrated in Figure 8.7(a), while the second setting requiring shifting the zero for a second time is shown in Figure 8.7(b).

The use of a second program zero is also applied to milling operations. An initial program zero is established and some machine movements will be made in relation to that datum. Then, via an appropriate program call the zero will be reestablished and all subsequent moves will be made in relation to the second datum.

One milling situation where the zero shift facility is particularly useful is when more than one component is to be machined at one setting, as illustrated in Figure 8.8. In this example a grid plate is used as a work-holding device. The advantage of the grid plate is that all the clamping and location points can be identified using a letter/number grid reference, like using a map reference

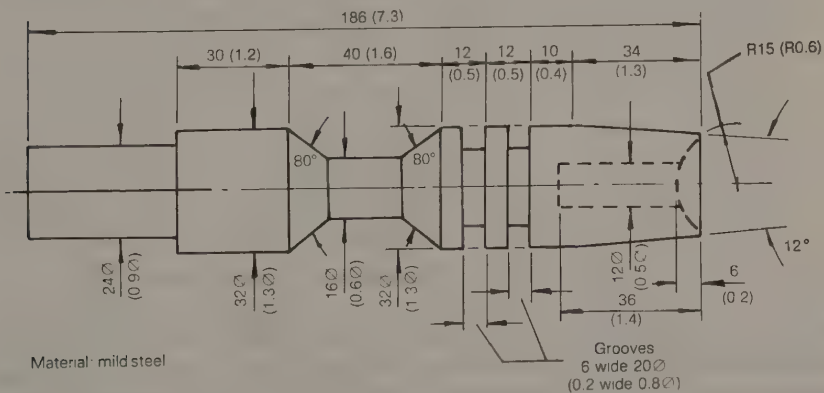


Figure 8.6 Component detail. (Inch units are given in parentheses.)

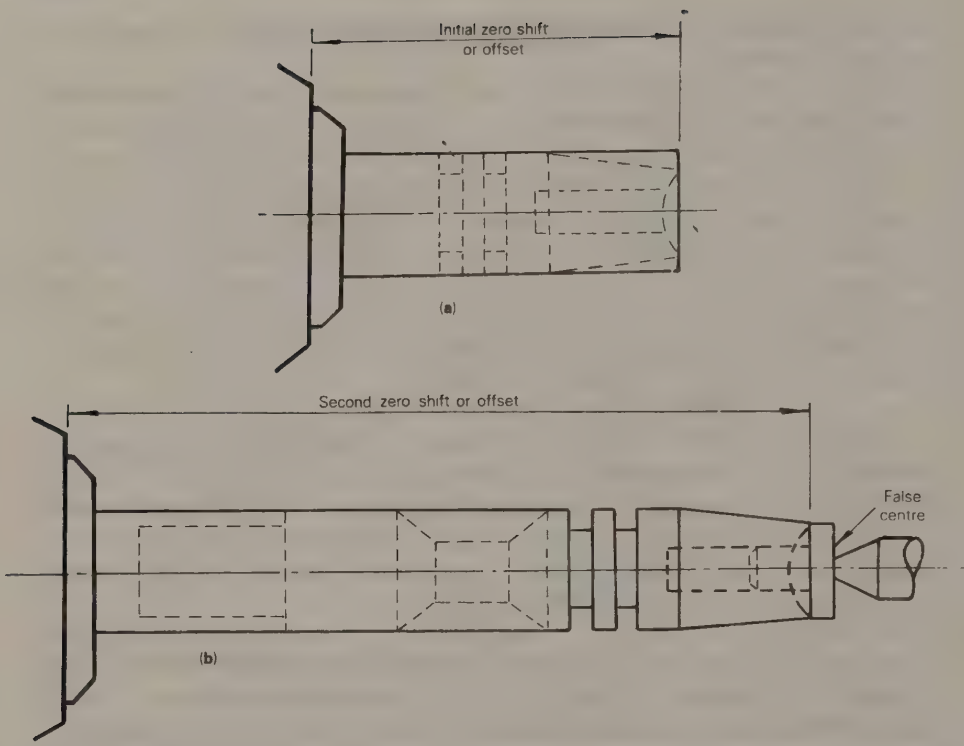


Figure 8.7 The application of a second zero shift to accommodate work resetting (a) first work setting (b) second work setting.

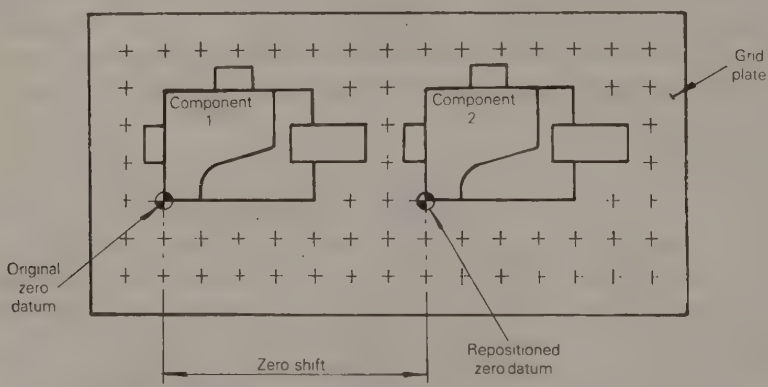


Figure 8.8 Use of zero shift for multicomponent machining at one setting.

to locate a particular town. Using this reference system the part programmer can instruct the machine set-up person/operator exactly where to position each component so that their location will correspond with the selected program zeros.

importance. The important thing is that the shop floor personnel fully understand what is required. So how detailed does the information need to be? The answer depends on the complexity of the component and the machining operations involved.

Assume that the machine set-up person knows the sequence of machining operations involved and is to proceed with setting up the machine. Consider in the first instance work loading, holding, and location. What information is required?

A simple component that is to be turned in one set up from a prefaced billet could be accommodated with a few short notes as follows:

Material:	prepared billet, part number ****
Loading:	manual
Work-holding:	chuck type, fixture number ****
Location:	back face of chuck
Zero shift:	Z direction + or -, and value

The last item would indicate that a manual data entry shifting the Z axis zero from the spindle face to the workpiece face is required.

A more complex component requiring two settings, with the second operation requiring center support activated by an entry in the program, will require a little more detail and the information may be given as follows:

Material:	diameter and length of bar stock
Loading:	bar feed to programmed stops, bar stop number
Work-holding:	collet, with programmed center support for second setting
Zero shifts:	first setting, direction + or -, and value second setting, direction + or -, and value

This information could be supplemented by two simple sketches showing the machining to be carried out at each setting.

A similar exercise can be carried out for workpieces involving milling. The exercise shown in Figure 8.10 could be produced on a "one part" basis or involve a multicomponent setting.

In the first instance the workpiece could be located using the corner of the fixed jaw of the vise as a reference point, a technique referred to on page 168. The instructions necessary to achieve this would be as follows:

Material:	prepared blank length $\times$ width $\times$ height
Work-holding:	machine vise, fixture number ****
Location:	left-corner of fixed jaw
Program datum:	X axis -25 mm (-1 in.) (axis-direction + or - value) Y axis 25 mm (1 in.) Z axis 2 mm (0.1 in.)

Again the information regarding the program datum may be more readily understood if the instructions include a sketch.

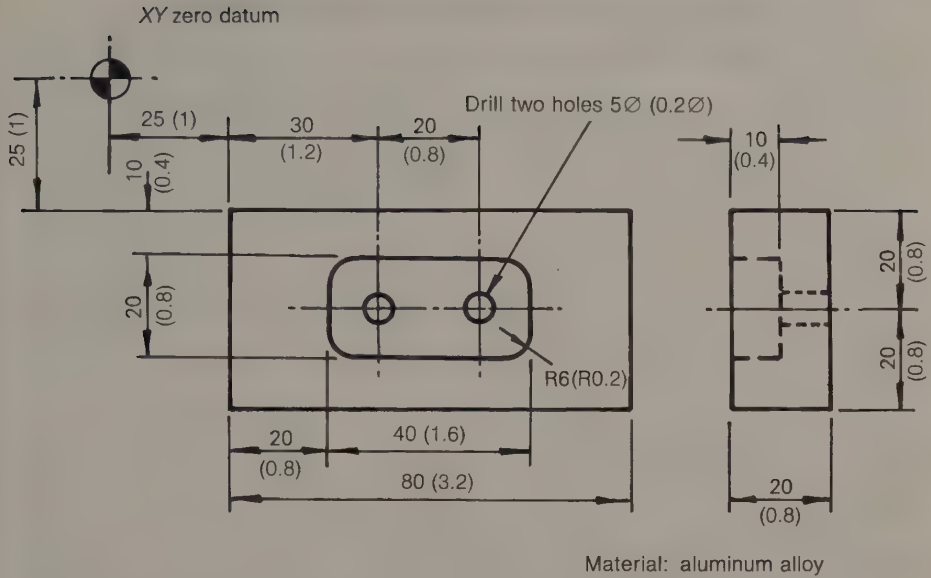


Figure 8.10 Component detail. (Inch units are given in parentheses.)

A multicomponent setup involving the same component could involve the use of a grid plate. To convey the necessary set-up information, the programmer should be familiar with the grid plate and its associated locating and clamping devices. With such knowledge he or she may be able to give detailed instructions for the complete setting, using the grid references to position the various setting blocks, locating dowels, and clamps to be used in the operation. On the other hand, a competent set-up person could manage with the basic information included in Figure 8.11.

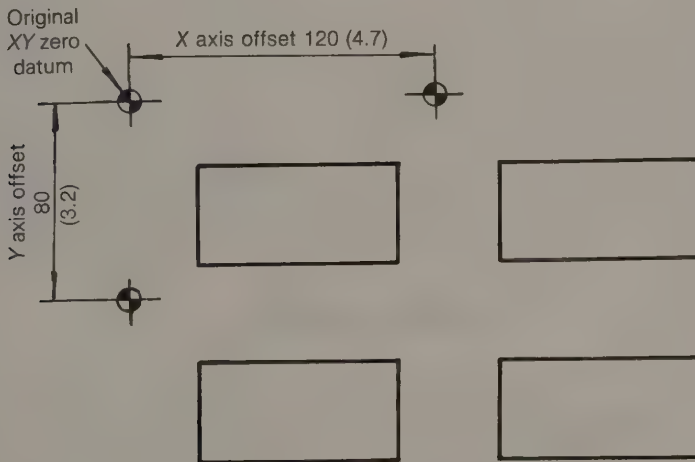


Figure 8.11 Use of grid plate. (Inch units are given in parentheses.)

TOOLING SELECTION AND IDENTIFICATION

The responsibilities of the part programmer concerning tooling are as follows:

- (a) determine the appropriate tool to be used, including their shape and size and the material from which they will be made;
- (b) allocate identification numbers to facilitate machine setting;
- (c) allocate tool offset numbers;
- (d) determine, when appropriate, the dimensional value of the offsets;
- (e) prepare appropriate documentation.

It is essential that a programmer is fully conversant with the tooling system for the machine involved, that is, the type of tooling that can be used and the way the tools can be located and held in position.

A major feature of CNC machining is the use of standard tooling. The intricate slide movements that are possible greatly minimize the need for special tooling, particularly form tools. In many ways the tooling requirements for CNC machining are less complex than for conventional machining.

Providing the programmer is conversant with the machine tooling system, the process of selecting tooling for a particular job is largely a case of selecting and utilizing standard items.

It is important that the correct tool material is used, particularly when using carbide inserts. Reference should be made to manufacturers' literature for guidance in this respect. Pages 44 through 46 give an indication of the type of information that is available.

It is often the case that the tools available within a company for use on a particular machine will be further standardized with their details being documented. An example of a company-based tool standard is shown in Figure 8.12.

All tools are required to have a numerical identity within the part program. This identity, commonly the letter T followed by two digits, is allocated by the part programmer and will correspond with the numbered position the tool will occupy in the machine turret, magazine, or other storage facility. The position each tool will occupy is affected by factors which are discussed below.

Commonly used tools are often given an identity that is retained at all times, since this often eliminates the need to reset when jobs are changed. When this situation exists, it is essential that the part programmer knows exactly which tools are involved and their numerical identity.

TOOL STORAGE

With automatic tool changing facilities involving turrets, the positions for the tools in the turret are numbered. Thus a tool call of, say, T06 will cause the turret to index to position number six. The tool allocated the numerical identity 6 must be set in position six.

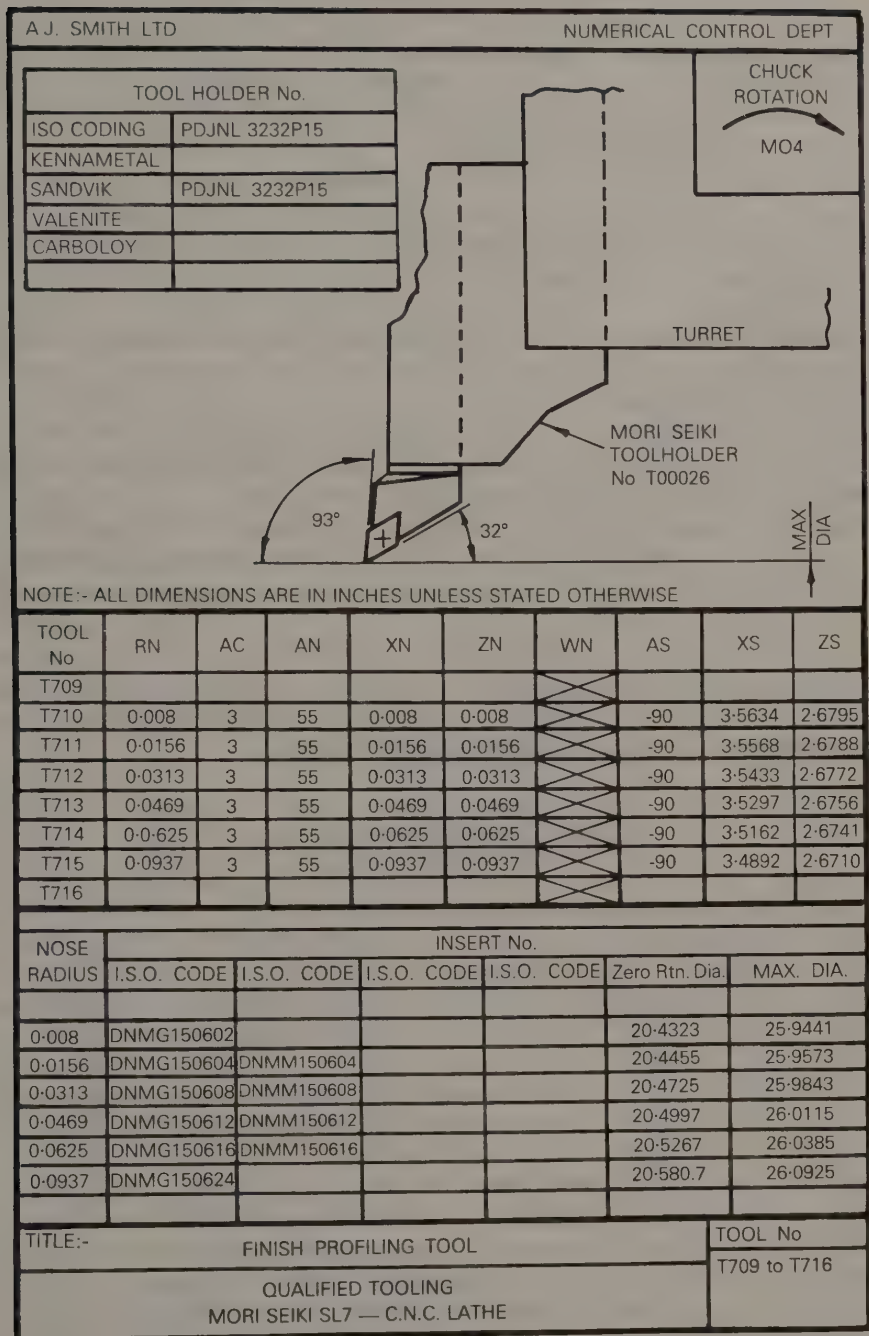


Figure 8.12 Company-devised tool standard. (All units are given in millimeters.)

Similarly, tools changed by automatic handling devices will be housed in readiness in a tooling magazine. When a tool is called the magazine will index to bring the appropriate tooling station into a position where the tool located in that station can be accessed by the handling device. Clearly the correct tool must be in each numbered position if the programmed tool call is to bring the desired tool into the machining position.

Even when the tool change is a manual operation, effected by a programmed stop in the machining cycle, the process is assisted if the operator has a clear indication of the next tool to be used. It is usual, therefore, to number the tool storage positions or even the tools themselves. When the programmed break in machining occurs, the operator can refer to a document provided by the programmer to determine the next tool involved; on the more sophisticated control systems the tool may be indicated by a message displayed on the visual display unit of the control.

The programmer should give due thought to the positioning of the tools in relationship to each other in the turret or magazine. Most indexing arrangements involve rotation in one direction only, so to change, say, from T03 to T06 will require three indexing moves, two of which are time-consuming and unproductive. Therefore the objective should be to position the tools in the turret or magazine in the order in which they will be called into use, although this is not always possible in practice.

The problem of wasteful indexing time is considerably eased when the machine is equipped with the facility to index tooling by the shortest possible route. In other words, the turret or magazine will rotate either clockwise or counterclockwise depending on which tool is called.

TOOL CHANGING POSITION

The programmer should consider carefully the position the machine slides are to be in when a tool change is made. There is a tendency, particularly among students, to return the machine slides to a set position before making a change, a practice that may have its merits from a safety point of view early in training but which, like wasteful indexing moves, can add considerably to the total time taken to machine the part.

The objective must be to keep noncutting slide movement to a minimum. For example, on a vertical machining center it is often possible to effect a tool change immediately above the point at which the tool completes the required machining, the change being carried out after an appropriate Z up-movement of the machine spindle or head. This saves making a long and unnecessary journey to a set position such as the XY zero datum. On turning centers a similar time saving can be achieved by indexing as near to the workpiece as is safely possible. The programmer should always refer to the machinery manuals to check clearances necessary to allow tool indexing mechanisms and cutting tools to clear any obstructions.

REPLACEMENT TOOLING

For long production runs the programmer will need to give some thought to the provision of replacement tooling.

When tools need to be replaced it is possible for the set-up person to determine suitable offsets and make the necessary tool data entries as he or she would for the original tools, but this is time-consuming and interrupts production.

An alternative approach is to use replacement tooling which is identical to the original. Such identical tooling may be of two types, namely, "qualified" or "preset."

Qualified tooling is used on turning centers and has dimensions guaranteed by the manufacturer to within ± 0.0005 in. or ± 0.08 mm from up to three datum faces.

Preset tooling is precisely set to predetermined dimensions in the toolroom and is applied to turning tools and milling cutters.

The programmer may choose to recommend qualified or preset tools when compiling his or her tooling schedule, but if such tooling is prescribed, the programmer may need a feedback of information from the toolroom regarding the setting sizes. This information then becomes part of the overall programming and machine-setting package and should be documented for future reference.

TOOLING DOCUMENTATION

Documentation regarding tooling, as with machine setting instructions, may be simple or relatively complex. It depends largely on the size of the company and the degree of organization that exists.

The possibilities range from the situation where the machine set-up person has personal access to the range of tooling likely to be required, to situations where the tooling is prepared in a special-purpose tool room, issued to the set-up person as a package for that particular job, and on completion returned to the tool room for refurbishment and storage.

For each programmed tool the minimum information required on the shop floor is as follows:

- (a) programmed identity—T01, T02, T03, etc.;
- (b) tool type;
- (c) holder type and size;
- (d) insert type and size;
- (e) overall dimensions (solid tools);
- (f) projection of cutting tool from holder.

When presetting is involved, the tool design or program personnel usually determine the original preset dimensions. The sizes should ultimately be no-

tified to the part programmer so that they may be recorded and included as part of the general documentation for that particular job. A well-organized tool preparation facility may well retain the data against their own job reference to facilitate the preparation of replacement tooling and to allow for the possibility of having to prepare identical tools at some future time. See Figures 8.13 and 8.14 for examples of tool data sheets.

When tooling offsets are being used to achieve a particular machining effect, as discussed on page 225, the value of the offsets must be included on the document.

It is often the situation that information regarding tooling, and sometimes information relating to machine setting, is included on the original part program form when one is used. Information documented in this way is of necessity rather brief, but in many cases is adequate.

Another practice widely adopted is to give tooling details alongside the tool call in the part program. Again, the information is brief but adequate for many situations.

The important thing is that the part programmer fully appreciates the needs of the people more directly concerned with the machining operation. There must be an efficient transfer of the relevant information. The means adopted

MODEL 104 TOOL SHEET											
PART NO. 001-001			MATERIAL 1018 CRS			OPERATION 3			PROGRAMMED BY G. COMBS		
PART NAME SAMPLE			B/P CHANGE DATE			DATE 5-6-80			SHEET 1 OF 1		
TOOL		TOOL DESCRIPTION	TOOL DIAMETER			TOOL LENGTH			OPERATION DESCRIPTION	SPEED	FEED
SEQ	NUMBER		PROG	ACT.	#	PROG	ACT.	#		R P M	IN/MIN
1	T01	HSS END MILL - 4 FLUTE 3/8" SHANK SINGLE END/CV-49- 15920 END MILL HOLDER			D1 1.000			H1	FINTSY MILL PERIPHERY		
	E1		1.0	.750		6.187	5.812			270	3
2	T02	HSS END MILL - 4 FLUTE 3/8" SHANK DBLE END/CV-49-15015			D2			H2	Mill (4) pockets .062 Dp. Finish mill sides		
	E1	END MILL HOLDER	.250	.250		4.687	4.687			1100	2
3	T03	#2 CENTER DRILL Erikson Ext Collet Chuck/#200-3/16 collet/CV-49-15923 Collet							SPOT (18) HOLES ON PERIMETER, (8) ON B.C. and (4) on angle.		
	E1	Holder/#100-3/4 collet						H3 .125	.1 DP		
			.078	.078		4.875	5.000			1180	3
4	T04	Drill CV-49-15923 collet holder/#100-1/8 collet						H4 -.250	Drill (8) HOLES ON B.C.		
	E1		.125	.125		5.125	4.875			3000	6
5	T05	DRILL CV-49-15923 collet holder/with #100-7/32 collet						H5 .250	DRILL (18) HOLES ON PERIMETER AND (4) ON ANGLE 1" THRU		
	E1		.205	.205		5.250	5.500			1650	5
6	T06	HSS END MILL - 2 FLUTE 3/8" SHANK DBL END/CV-49-15915 END MILL HOLDER						H6 .188	C'BORE (18) HOLES		
	E1		.375	.375		4.562	4.700			850	3.4

Figure 8.13 Model 104 tool sheet. Ex-Cell-O Corp., Rockford Machine Tool Company, Rockford, IL

to achieve this objective will vary, but the programmer should always remember that it is a very important aspect of his or her work.

PART PROGRAMMING PROCEDURE

The blocks of data entered in a part program are numbered N01, N02, N03, and so on. On completion of a machining program it is usually necessary to return to the beginning so that another component can be machined. The return to the program start position is usually achieved via a "rewind" or "return to start" command included at the end of the program.

With word address systems, this command is entered as a miscellaneous function designated M30, which has the effect of stopping all slide and spindle movement, turning off the coolant supply and rewinding the tape. When the tape has merely been used to transfer a program into the microcomputer memory, then it rewinds the program within the microcomputer. The stage at which this rewind must cease has to be identified, and this is achieved via a "rewind stop" program entry signified by the % sign. This is usually the first entry in a word address program.

With the start of the program established, the next three or four blocks of data will concern setting the machine controller so that it interprets subsequent data in the correct manner. These set-up entries include instructions relating to the following:

- (a) *units*, which may be programmed in inch or metric;
- (b) *slide movement*, which may be stated as incremental or absolute dimensional values;
- (c) *speed*, which may be programmed as surface speed in feet/meters per minute or spindle speed in revolutions per minute;
- (d) *feed*, which may be programmed as inches/millimeters per spindle revolution or inches/millimeters per minute.

Having established the basic set-up data, it may be helpful now to list in a general way the functions and machine movements necessary to produce the component. Consider the drawing for Exercise 1 (in Appendix C) and imagine that the machine is set with the spindle in its 'home' or 'base' datum position, that is, at a point some distance above the XY datum indicated on the drawing. Starting from this position, the part program must provide for the following:

1. Rapid linear movement to P1 in X and Y.
2. Rapid linear movement to a clearance position above Z0.

3. Spindle on clockwise direction.
4. Coolant on.
5. Feed linear movement to Z depth.
6. Rapid linear movement to clearance above Z0.
7. Rapid linear movement to P2.
8. Feed linear movement to Z depth.
9. Rapid linear movement to clearance above Z0.
- . . . and so on.

These simple comments can, providing a space exists, be entered directly onto a program sheet or, if the program is being listed on plain paper, alongside each item of data, but it is probably a better plan to prepare a rough list in the first instance and then check carefully to ensure nothing has been overlooked. Relative codes and data can then be added to each statement.

Should it be found that, on completion of a program, omissions have inadvertently been made, the error can be rectified more easily if the block numbers are allocated in increments of five: N01, N05, N10, N15. It is then a simple matter to include additional blocks—N06, N07, N08, for instance—between N05 and N10.

If the program is being listed on a computer the blocks can be numbered consecutively, since any omission entered via the keyboard will automatically cause the existing blocks to renumber or, alternatively, renumbering can be easily effected. Many MDI control systems also have this facility.

A methodical approach to part programming is essential, and it is recommended that, even for a simple component, an operation schedule listing the tooling speeds and feeds to be used should be completed in the first instance.

WORD ADDRESS PROGRAMMING

Word address programming is largely based on an International Standards Organization (ISO) and Electronic Industries Association (EIA) code that require the program to be compiled using codes identified by letters, in particular G and M. Each code addresses, or directs, the item of data it precedes to perform a certain function within the control system.

The ISO and EIA Standards provided for 99 G codes and an identical number of M codes, each being expressed by the address letter followed by two digits.

Not all the codes were allocated a specific function in the Standard and this gave the manufacturers of control systems the opportunity to introduce their own variations. There is, therefore, no standard word address machine programming language, although many of the recommendations made have been widely adopted.

The G codes, or preparatory functions, are used to set up the machine control unit modes of operation required for the machining that is to be carried out—whether movement is to be in a straight line/linear or radially/circular, for example. In general they relate to slide motion control. Examples of commonly used G codes are as follows:

G00	Rapid linear positioning, point to point
G01	Linear positioning at a controlled feed rate
G02	Circular interpolation, clockwise
G03	Circular interpolation, counter-clockwise
G04	Dwell for programmed duration
G33	Thread cutting, constant lead
G34	Thread cutting, increasing lead
G40	Cutter compensation, cancel
G41	Cutter compensation, left
G42	Cutter compensation, right
G70	Inch programming
G71	Metric programming
G80	Series associated with drilling, boring, tapping and reaming.

(For a complete list of G codes refer to Chapter 6.)

G codes may be “modal,” that is, they remain active until cancelled. Alternatively they may be nonmodal, and are only operative for the block in which they are programmed.

The M codes, or miscellaneous functions, are used to establish requirements other than those related to slide movement. For example, they are used to activate spindle motion or to turn on a coolant supply. Examples of commonly used M codes are as follows:

M00	Program stop
M01	Optional stop
M02	End of program
M03	Spindle on clockwise
M04	Spindle on counter-clockwise
M05	Spindle off
M06	Tool change
M08	Coolant on
M09	Coolant off
M30	End of tape

(For a complete list of M functions refer to Chapter 6.)

As with G codes, some M functions are modal, remaining active until cancelled. M functions may also become active immediately upon reading of the

block or after all block commands are completed. (Refer to machinery manuals to determine how various codes operate.)

In addition to the address letters G and M there is also common usage of S, F, and T to indicate speeds, feeds, and tooling. The letter N is always used to identify block numbers.

The distinction between word address and conversational programming is best appreciated by reference to the simple movements discussed earlier.

To program the linear movement of -39.786 mm or -1.6 in. in the X axis using the word address technique, it is first necessary to establish the operating mode required. This is done by including the appropriate G code, in this case G01. Thus the complete program entry for the required move will be:

Inch N260 G01 X-1.6

Metric N260 G01 X-39.786

Similarly, reconsider the 0.3 in. or 8 mm radial movement through an arc of 90° . Once again the mode of operation has to be established using the appropriate G code, which for circular movement in a clockwise direction is G02. It will also be necessary to define the target position in the appropriate axes and also the start of the arc in relation to the arc center using I, J, and K address letters that correspond to the X , Y , and Z axes respectively. A word address program entry to achieve this movement would read as follows:

Inch N350 X1.7 Z-3.0 K.3

Metric N350 G02 X43.765 Z-75.000 K8

There are variations in procedure even when word address programming such a common machining feature as a radius. On some control systems the arc center may have to be defined—still using the I, J, and K address letters—in relation to the program datum and not the start position.

The programming of radial movements using the word address method will be returned to later in the text.

A word address program that includes a number of codes in both inch and metric is listed below. The program relates to the component detailed in Figure 8.15, and is typical of its type. The comments written alongside the data should convey to the reader an impression of how, prior to programming, the machining of a component is first broken down into operations. It also shows how the necessary machine control data are presented. Later in the text further reference will be made to the program to illustrate specific programming techniques and features.

PROGRAMMING EXAMPLE

Figure 8.15 shows a simple turned component for which a part program is to be prepared using the following basic programming information. (Examples of more detailed programming specifications are given in Appendix C.)

PREPARATORY FUNCTIONS (G CODES)

- G00 Rapid movement
- G01 Linear interpolation—movement at a programmed feed rate
- G02 Circular interpolation, clockwise
- G03 Circular interpolation, counter clockwise
- G40 Cancel tool nose radius compensation
- G41 Tool nose radius compensation left
- G42 Tool nose radius compensation right
- G70 Inch units
- G71 Metric units

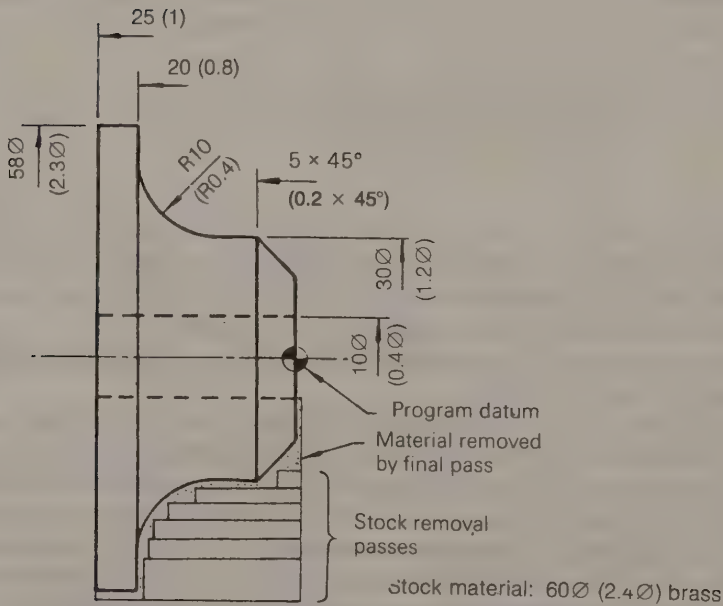


Figure 8.15 Component detail. (Inch units are given in parentheses.)

G90	Absolute positioning data	} X axis values to be programmed as diameters
G91	Incremental positioning data	
G94	Feed (in. or mm)/min	
G95	Feed (in. or mm)/rev	
G96	Constant surface cutting speed	
G97	spindle speed rev/min	

MISCELLANEOUS FUNCTIONS (M CODES)

M00	Program stop
M01	Optional stop
M02	End program
M03	Spindle on clockwise
M04	Spindle on counterclockwise
M05	Spindle off
M06	Tool change
M08	Coolant on
M09	Coolant off
M30	End of program

OPERATION SCHEDULE

The first stage in the programming process is to prepare an operation schedule. An operation schedule for the component is shown in Figure 8.16, where only metric units are shown. The spindle speeds and feed rates have been determined by reference to the cutting data given in Chapter 7.

TOOLING INFORMATION

Although the component is a relatively simple one, it is still necessary to provide tooling information for the machine tool setter. This information is detailed on the form illustrated in Figure 8.17, where only metric units are shown.

PROGRAM LISTING

Attention can now be given to listing the necessary programming data, together with appropriate remarks to ensure a logical approach is being adopted and to

OPERATION SCHEDULE		PART No. <i>EX. 1</i>	DESCRIPTION <i>PLUG</i>		SHEET No. <i>1 of 1</i>	
		MACHINE TYPE <i>H8370</i>	COMPILED BY <i>A.R.C.</i>		DATE <i>9-2-84</i>	
OP No.	DESCRIPTION	TOOLING TYPE AND SIZE	WORK HOLDING	CUTTING SPEED	FEED RATE	SPINDLE SPEED
1	<i>CENTRE DRILL</i>	<i>HSS No 2 C/DRILL</i>	<i>φ60</i>	<i>28</i>	<i>.12</i>	<i>1500</i>
2	<i>DRILL</i>	<i>HSS DRILL φ10</i>		<i>28</i>	<i>.18</i>	<i>890</i>
3	<i>TURN PROFILE</i>	<i>CEM. CARB. INSERT</i>		<i>170</i>	<i>.25</i>	<i>1350</i>
4	<i>PART OFF</i>	<i>CEM. CARB. INSERT</i>		<i>170</i>	<i>.16</i>	<i>1350</i>

Figure 8.16 (Metric units)

TOOL PREPARATION AND SETTING DATA				PART No. <i>EX. 1</i>		
TURRET POSITION	OFFSET No.	OPERATION	INSERT TYPE	HOLDER TYPE	PRE SET LENGTHS	
					X	Z
1	01	<i>CENTRE DRILL</i>	<i>—</i>	<i>RC 107</i>	<i>DETERMINE AND ENTER OFFSETS ON THE MACHINE</i>	
3	03	<i>DRILL φ 10</i>	<i>—</i>	<i>RC 110</i>		
4	04	<i>TURN TO PROFILE</i>	<i>P10</i>	<i>TN 22-08</i>		
5	05	<i>PART OFF</i>	<i>P20</i>	<i>GR 18-04</i>		

Figure 8.17 (Metric units)

ensure that nothing is overlooked. The required program is listed below. (Note that, in this particular case, a programming form is not being used, but partially completed programming exercises involving the use of a form are given in Appendix C.)

PART PROGRAM (INCH)

Data	Remarks
N10 G70 G90	Absolute inch
N15 G95 G97	Feed inches/rev Spindle speed rev/min
N20 G92 X4.0 Z8.0	Pre-set safe turret indexing position
N25 T0101 M06	Tool change. Tool No. 1. Off-set No. 1
N30 S3000 M03	Spindle on clockwise
N35 G00 X0 Z.1 M08	Rapid to start position. Coolant on
N40 G01 Z-.3 F.003	Center drill
N45 G00 Z.1	Rapid retract
N50 X4.0 Z8.0	Return to turret index position
N55 T0202 M06	Tool change. Tool No. 2 Off-set No. 2
N60 S2380 M03	Spindle Speed
N65 G00 X0 Z.1	Rapid to start position
N70 G01 Z-1.2 F.015 M08	Drill through .4ø
N75 G00 Z.1	Rapid retract
N80 X4.0 Z8.0	Return to turret index position
N85 T0303 M06	Tool change. Tool No. 3 Off-set No. 3
N90 S1430 M03	Spindle speed
N95 G00 X1.97 Z.1	Rapid to start position
N100 G01 Z-.768 F.020 M08	
N105 G00 X2.05 Z-.688	Rapid retract to clear cut surface
N110 Z.1	Second rough pass—start position
N115 X1.772	
N120 G01 Z-.748	
N125 G00 X1.85 Z-.669	
N130 Z.1	
N135 X1.575	Third rough pass—start position
N140 G01 Z-.709	
N145 G00 X1.654 Z-.63	
N150 Z.1	
N155 X1.417	Fourth rough pass—start position
N160 G01 Z-.65	
N165 G00 X1.496 Z-.57	
N170 Z.1	
N175 X1.26	Fifth rough pass—start position
N180 G01 Z-.512	
N185 G00 X1.339 Z-.433	
N190 Z.1	
N195 X.984	Sixth rough pass—start position
N200 G01 Z-.079	
N205 G00 X1.063 Z.1	

Data		Remarks	
N210	G41	Cutter radius compensations	} Finish machine profile
N215	X0	Rapid to X zero	
N220	G01 Z0	Rapid to Z zero	
N225	S1750 M03	Spindle speed and feed rate change	
N230	X1.0 F.007	Machine face to 1.0ø	
N235	X1.2 Z-.2	Machine chamfer	
N240	Z-.4	Linear move to radius start	
N245	G03 X2.0 Z.8 I.4	Circular interpolation	
N250	G01 X2.3	Linear move to 2.3ø	
N255	Z-1.1	Linear move to length	
N260	G00 X2.5	Lift from finished surface	
N265	G40	Cancel cutter radius compensation	
N270	X4.0 Z8.0	Return to turret index position	
N275	T0404 M06	Tool change. Tool No. 4 Parting Tool	
N280	S1430 F.007	Spindle speed and feed rate change	
N285	G00 X2.5 Z-1	Rapid to start	
N290	G01 X.08 M08	Part off leaving stock faced	
N295	G00 X4.0 Z8.0	Return to turret index position	
N300	G92 X0 Z0 M30	Program end. Spindle and coolant off	

Note: To simplify the program, neither tool nose radius or thickness were used.

PART PROGRAM (METRIC)

Data		Remarks
N10	G71 G90	Absolute metric
N15	G95 G97	Feed mm/rev Spindle speed rev/min
N20	G92 X100 Z200	Pre-set safe turret indexing position
N25	T0101 M06	Tool change. Tool No. 1. Off-set No. 1
N30	S3000 M03	Spindle on clockwise.
N35	G00 X0 Z2 M08	Rapid to start position, coolant on
N40	G01 Z-8 F.1	Center drill
N45	G00 Z2	Rapid retract
N50	X100 Z200	Return to turret index position
N55	T0202 M06	Tool change. Tool No. 2 Off-set No. 2
N60	S2380 M03	Spindle speed
N65	G00 X0 Z2	Rapid to start position
N70	G01 Z-30 F.18 M08	Drill through 10ø
N75	G00 Z2	Rapid retract
N80	X100 Z200	Return to turret index position
N85	T0303 M06	Tool change. Tool No. 3 Off-set No. 3
N90	S1430 M03	Spindle speed
N95	G00 X50 Z2	Rapid to start position
N100	G01 Z-19.5 F.3 M08	
N105	G00 X52 Z-17.5	Rapid retract to clear cut surface
N110	Z2	
N115	X45	Second rough pass—start position

Data	Remarks	
N120 G01 Z-19		
N125 G00 X47 Z-17		
N130 Z2		
N135 X40	Third rough pass—start position	
N140 G01 Z-18		
N145 G00 X42 Z-16		
N150 Z2		
N155 X36	Fourth rough pass—start position	
N160 G01 Z-16.5		
N165 G00 X38 Z-14.5		
N170 Z2		
N175 X32	Fifth rough pass—start position	
N180 G01 Z-13		
N185 G00 X34 Z-11		
N190 Z2		
N195 X25	Sixth rough pass—start position	
N200 G01 Z-2		
N205 G00 X27 Z2		
N210 G41	Cutter radius compensations	} Finish machine profile
N215 X0	Rapid to X zero	
N220 G01 Z0	Rapid to Z zero	
N225 S1750 M03	Spindle speed	
N230 X20 F.15	Machine face to 20ø	
N235 X30 Z-5	Machine chamfer	
N240 Z-10	Linear move to radius start	
N245 G03 X50 Z-20 I10	Circular interpolation	
N250 G01 X58	Linear move to 58ø	
N255 Z-26	Linear move to length	
N260 G00 X64	Lift from finished surface	
N265 G40	Cancel cutter radius compensation	
N270 X100 Z200	Return to turret index position	
N275 T0404 M06	Tool change. Tool No. 4 parting tool	
N280 S1430 F.18	Spindle speed and feed rate change	
N285 G00 X64 Z-25	Rapid to start	
N290 G01 X-2	Part off leaving stock faced	
N295 G00 X100 Z200	Return to turret index position	
N300 G92 X0 Z0 M30	Program end. Spindle and coolant off	

Note: To simplify the program no tool nose radius or thickness was used.

DATA FORMAT

Data are written in blocks. The data within a block were once expressed in a fixed sequence with each block containing all data (even if they have not changed from the previous block), but now almost exclusively the commands appear in random order without the repetition of unchanged data but with each word being clearly identified by its address letter. The terminology used to describe these two methods is “fixed block” and “variable block word address,” respectively. The following examples illustrate these formats.

Fixed Block Example

N	G	X	Y	Z	I	J	K	F	S	M	Remarks
0250	00	05000	09000	04000	00000	00000	00000	2000	0350	03	Rapid position X, Y, Z, and start spindle.
0300	01	08500	09000	04000	00000	00000	00000	0310	0350	08	Mill in X axis—turn coolant on
0350	00	08500	09000	04500	00000	00000	00000	2000	0350	09	Retract Z axis—turn coolant off

Note: When information is placed in machine format, no spaces occur between data words.

Variable Block Example (spaced out in a form)

N	G	X	Y	Z	I	J	K	F	S	M	Remarks
N0250	G00	X05000	Y09000	Z04000	—	—	—	F2000	S0350	M03	Rapid position X, Y, Z, and start spindle
N0300	G01	X08500	—	—	—	—	—	F0310	—	M08	Mill in X axis—turn coolant on
N0350	G00	—	—	Z04500	—	—	—	F2000	—	M09	Retract Z axis—turn coolant off

Variable Block Example (without form—using decimal point format)

N0250	G00	X5	Y9	Z4	F200	S350	M03	Rapid position X, Y, Z, and start spindle.
N0300	G01	X8.5	F31	M08				Mill in X axis—turn coolant on.
N0350	G00	Z4.5	F200	M09				Retract Z axis—turn coolant off.

It is necessary for the part programmer to be aware of the data format for the system being used, and also to be familiar with the classification of the data that dictates the way in which it may be presented within a block. For example, a programming manual could indicate that data must conform to the following classification:

N4, G2, X3/3, Y3/3, Z3/3, F4, S4, T2, M2 (METRIC)
 N4, G2, X2/4, Y2/4, Z2/4, F4, S4, T2, M2 (INCH)

This classification indicates the following:

N4	The block sequence address letter N may be followed by up to four digits.
G2	The preparatory function address letter G may be followed by up to two digits.
X3/3, Y3/3, Z3/3 (Metric) X2/4, Y2/4, Z2/4 (Inch)	The axis identification letters X, Y, and Z may be followed by up to three digits in front of the decimal point, and up to three after in metric form. Identification letter may be followed by two digits in front of decimal point, and up to four after in inch format. (Dimensional values may be subject to other limitations as explained below.)
F4	The feed address letter F may be followed by up to four digits.
S4	The spindle speed or cutting speed address letter S may be followed by up to four digits.
T2	The tool address letter T may be followed by up to two digits.
M2	The miscellaneous function address letter M may be followed by up to two digits.

(Note: These formats will change from one machine control to another. For a complete list of your machine code formats refer to your machine tool manual.)

The description above has stated that up to so many digits may be used. Some systems require that leading zeros are included and some do not. Thus a linear slide movement at a programmed feed rate may be programmed as G01 or G1, depending on the system used.

Similarly, dimensional values may also have to be programmed according to certain rules. For instance, using a data classification of 3/3 it would be possible, depending on the requirements of the system, to program a value of 32 mm in a number of ways:

- 032000—all digits must be included but no decimal point.
- 32000—leading zeros are omitted, but no decimal point is required; trailing zeros must be included.
- 32.000—the decimal point and all trailing zeros are required.
- 32.—no leading or trailing zeros are required but the decimal point must be included.

- (e) 32—whole numbers may be programmed without leading or trailing zeros and without a decimal point.

SLIDE MOVEMENTS

Both word address and conversational programming require definition of the slide movements necessary to position the cutting tool correctly in relation to the work.

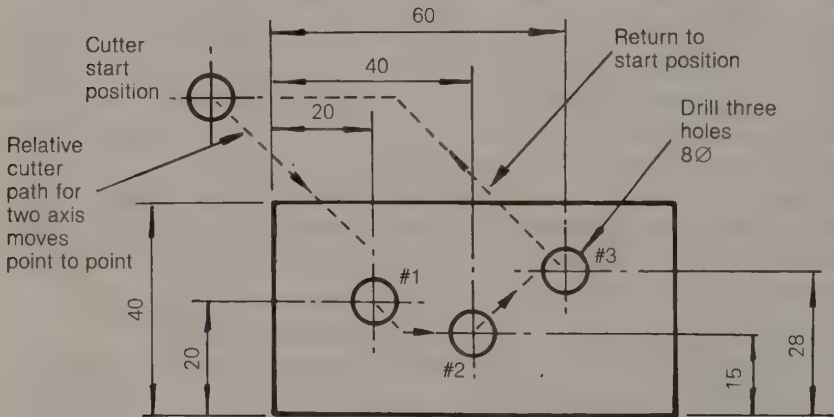
This positioning is described in three ways:

- (a) point-to-point;
- (b) linear interpolation;
- (c) contouring/circular interpolation.

Point-to-point positioning involves programming instructions that identify only the next relative tool position required. The position may be reached by movement in one or more axes at a rate of travel that is generally, though not necessarily, the maximum for the machine. If metal cutting takes place during this type of motion, it must be in one axis or the cut path will not necessarily repeat cycle to cycle.

Figure 8.18 shows details of a component. To drill the holes in this component would require two-axis point-to-point positioning. Note that the positioning prior to drilling is clear of obstruction, therefore path is not important and the actual drilling is a single-axis move, so point to point can be used. Note that point-to-point positioning is not capable of machining angles or contours, because they require controlled movement of more than one axis.

Linear interpolation control requires programmed instructions that specify



Material: mild steel 12 mm thick

Figure 8.18 Component detail requiring point-to-point positioning to drill holes.

both the next position and the rate of travel, or feed rate, to be employed to reach that position; the resulting cutter path is a straight line. Metal cutting would normally take place during such a move. Linear interpolation allows the machining of straight lines at a feed rate using one or two axes of motion. Other more expensive machines allow linear interpolation in three axes simultaneously. Figure 8.19 illustrates examples of one- and two-axis linear interpolation moves.

Contouring is used to describe movements involving at least two slides. The movements occur simultaneously and at a predetermined feed rate, and result in a continuous machining path which is not a straight line. An elliptical profile or a combination of arcs—the production of an arc being referred to as ‘circular interpolation’—are good examples of contouring. Contouring will normally refer to irregular curves that must be machined using minute straight line segments to generate it. Contouring requires many data blocks and multiple axis movement capability. Circular interpolation, on the other hand, will produce uniform arc segments or circles with minimal programming owing to the machine control’s ability to self-generate uniform arc data. The principle is illustrated in Figure 8.20.

DEFINITION OF THE AXES OF MOVEMENT

Whether conversational or word address programming is being used, the direction in which slide movement is to occur is defined by a letter, which for common machines is either X, Y, or Z for linear movement and B for rotary table movement where applicable. Axes definition codes appear together with

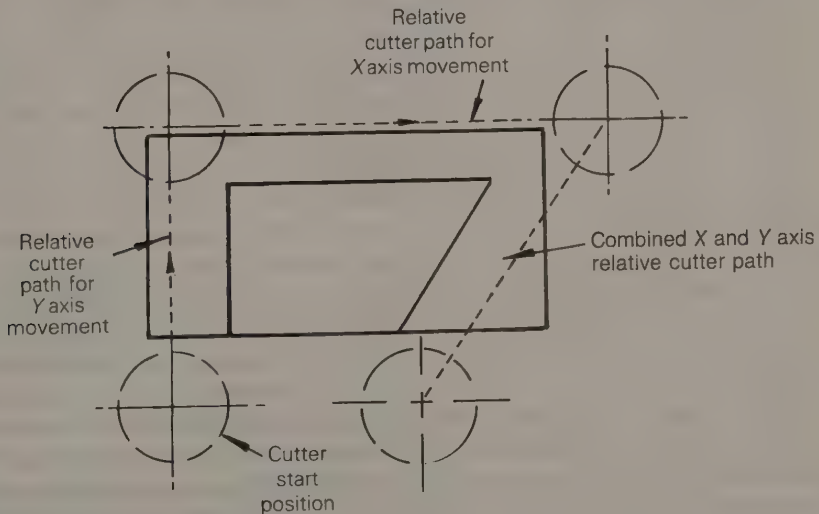


Figure 8.19 Linear interpolation.

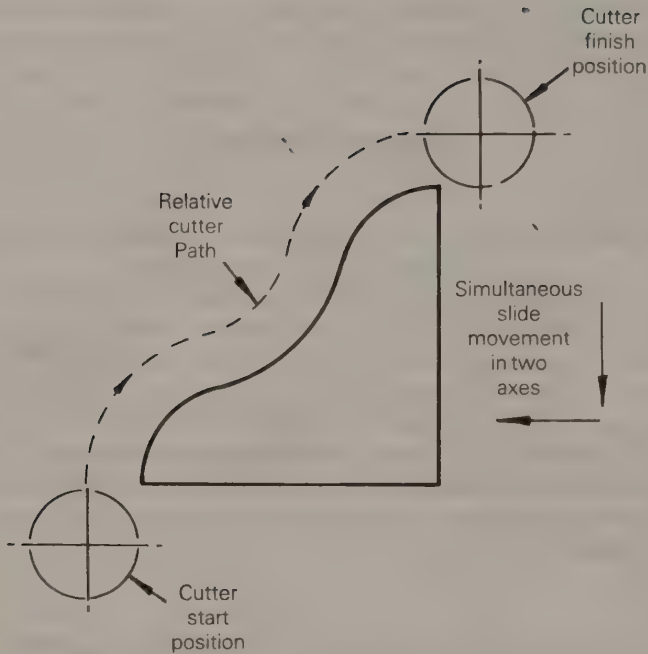


Figure 8.20 Contouring.

a positive (+) or negative (-) sign for determining direction. In practice the + sign is not actually entered, because if a sign is omitted, the control automatically assumes plus.

The definition of the axes of movement on common machine types, namely a turning center, a vertical machining center, and a horizontal machining center, are illustrated in Figure 8.21. Two points should be noted in relation to the illustrations. First, on a turning center having a rear-mounted tool post the plus (+) and minus (-) in the X axis would be reversed. Movement of the tool away from the spindle axis is always plus.

Second, the axes definitions shown indicate the *machine* slide movements. In the case of a turning center these movements are identical to the tool movement in relation to the work. On milling machines, where it is the table and not the cutting tool which moves, this is not the case. For programming purposes, where it is easier to imagine that the tool is moving, it is necessary to redefine some movements. On a vertical machining center, for example, in order to achieve a tool movement in relation to the workpiece in the X positive or plus direction it is necessary to build a machine slide that movement in the X axis would cause physical table movement to the right under the machine spindle viewed from the front of the machine. In determining axes directions, the programmer can always consider that the part is viewed through the tool from the shank to the tip. Right-theoretical tool motion on the part is always

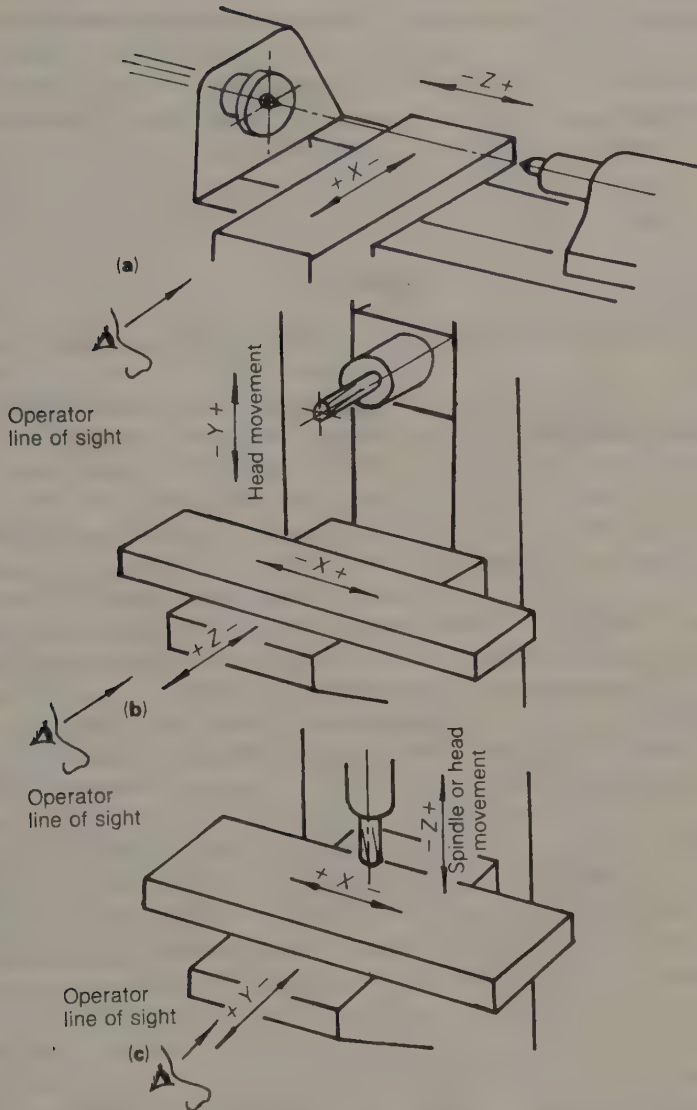


Figure 8.21 Identification of slides and direction of the slide movement on common machine tools: (a) center lathe (turning center), (b) horizontal milling machine (horizontal machining center), (c) vertical milling machine (vertical machining center).

positive, with left being negative. Tool motion viewed this way in the up direction on the part print is positive, with down being negative. Tool motion causing penetration of the tool into the work is negative, with retraction being positive. If rotary tables are involved, clockwise direction viewed looking into the table face from outside is positive, with counterclockwise being negative.

Note: these directions are based on a normal operator position in front of the machine.

In addition to linear movement, the production of a part may also require rotary movement which is provided by the use of ancillary equipment such as rotary tables and indexers. These movements are also controlled via the machining program and are identified by the letters A, B, and C as illustrated in Figure 8.22.

DATUMS

There are two datums involved in CNC machining that concern the part programmer.

The first of these is the program datum, which is established by the programmer when writing the program. This datum is at the intersection of the X, Y, and Z axes when milling, and at the intersection of the X and Z axes when turning. In both cases it is given the numerical identity of zero. The actual position of this datum in relation to the workpiece is optional, although there are certain factors to be taken into consideration.

The program datum is, in effect, the point from which the slide movements in each axis will be dimensionally related. It should be noted that part setup on a machine tool must allow for the proper machine zero/program zero relationship. When program data are stated in absolute terms (see below), all subsequent moves will also be dimensionally related to that point.

The second datum that concerns the part programmer is the machine datum. This datum is a set position for the machine slides where the axes intersect, and it has a numerical identity of zero within the control system.

On some machines the machine datum is a permanently established position (referred to as a fixed zero) and cannot be altered, although it can be repositioned on a temporary basis via a zero "shift" or "offset" facility. On other machines a new datum can be established anywhere within the operating pocket of the machine, a facility referred to as a "floating zero."

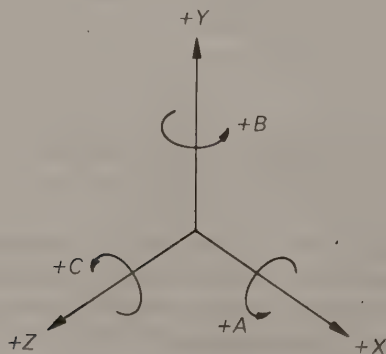


Figure 8.22 Identification of rotary movements.

Clearly, there must be some correlation between the machine datum and the program datum when setting the machine if the programmed slide movements are to achieve the intended effect, and the practicalities involved are discussed in more detail in Chapter 6.

ABSOLUTE AND INCREMENTAL POSITIONAL DATA

Once the direction of movement has been established, the distance moved by the machine slide to bring it to a desired position has to be defined dimensionally. This is achieved by the use of linear coordinates, with the dimensions being stated in absolute or incremental terms.

A third method is sometimes used, involving the use of polar coordinates. This requires a distance (the radius) stated in relation to a defined point and at an angle stated in relation to a datum axis. It generally requires the control system to include a special programming facility.

Absolute dimensional definition requires all slide movements to be related to a predetermined zero datum.

Incremental dimensional definition requires each slide movement to be related to the final position of the previous move.

Figure 8.23(a) shows the details of a turned component. The intersection of the spindle centerline and the face of the work is the program zero datum. Assume that a final trace of the component profile is to be programmed.

The dimensional definition in absolute and incremental values that would be required to define slide movement is shown in Figures 8.23(b) and 8.23(c) respectively.

In Figure 8.24(a) the details of a milled component are given. Absolute and incremental dimensional values required to program the machining of the slot are shown in Figures 8.24(b) and 8.24(c) respectively.

Earlier programming languages required dimensional data to be stated in *either* absolute or incremental terms. Modern controllers often provide a “mix and match” facility that permits the use of both within the same program, and even within the same data block. The distinction is achieved by the continued use of the G90 (absolute) and G91 (incremental) preparatory function codes, or X, Y, and Z word addresses for absolute values and U, V and W for incremental values. The use of the “G” code for changing from absolute to incremental is much more common.

CIRCULAR INTERPOLATION

Circular interpolation allows for programming a machine to move the cutting tool in a circular path with a uniform arc, with only a few commands. Circularlike motion is achieved through the machine controls capability to cal-

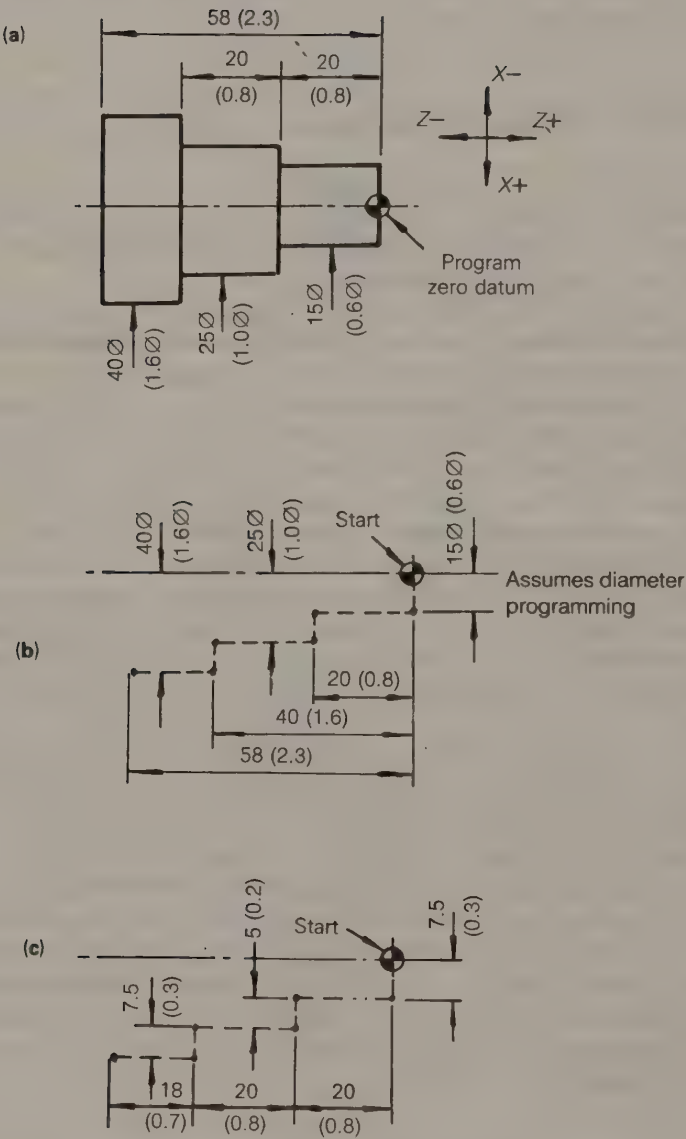


Figure 8.23 (a) Component detail, (b) absolute dimensions, and (c) incremental dimensions. (Inch units are given in parentheses.)

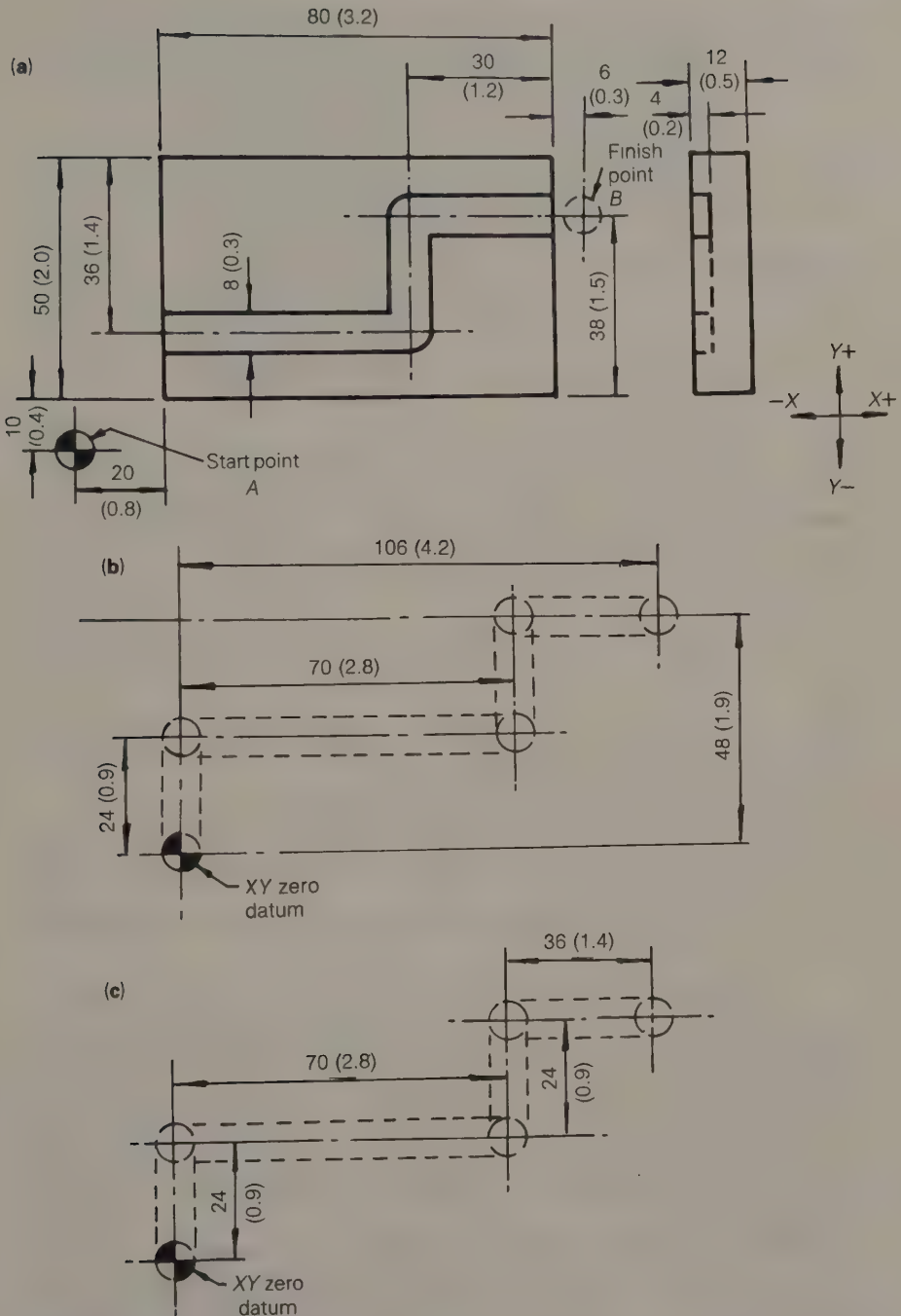


Figure 8.24 (a) Component detail, (b) absolute dimensions, and (c) incremental dimensions. (Inch units are given in parentheses.)

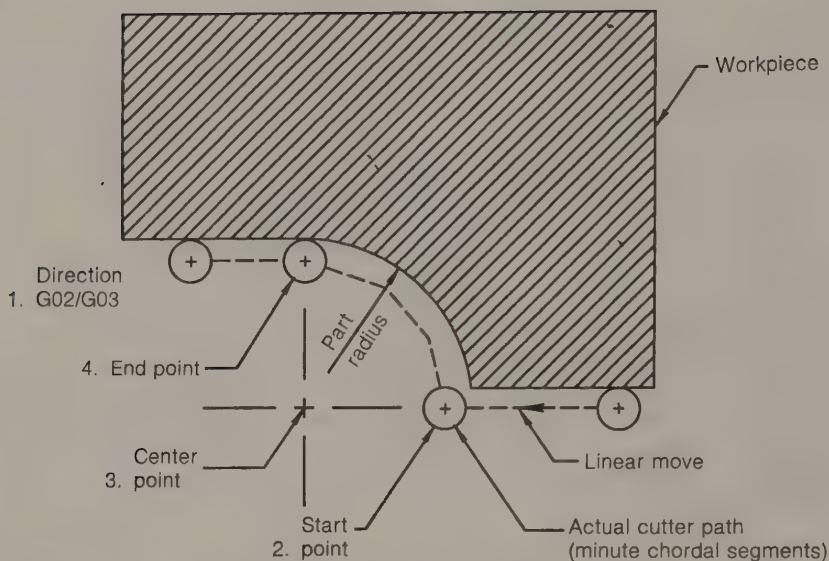


Figure 8.25 *The four basic elements of circular interpolation.*

culate minute cordal movements between commanded points (Figure 8.25). The basic elements of circular interpolation will deal with the following four items:

1. *Direction of rotation*—The direction of rotation that the cutter path will follow around the circle or arc segment.
2. *Start point*—The actual beginning point of the circle or arc segment where the cutter has previously been positioned.
3. *Center point*—The point of rotation in which the circle or arc segment is developed around.
4. *End point*—The finish point of the cutter path used to generate the circle or arc segment.

It was explained earlier in the text that circular arc programming on conversational MDI systems has been reduced to a simple data entry that specifies the target position, the value of the radius, and the direction of rotation. This simple method of defining circular movement in a single block with the direction of rotation being defined by the appropriate G code is also available on some word address systems (see method 3), but many systems employ one of two slightly more complex techniques of quadrant circular interpolation (methods 1 and 2).

Common to all programming systems is the need to determine whether the relative tool travel to produce a particular arc is in a clockwise (CW) "G02" or counterclockwise (CCW) "G03" direction, and the location of the arc or circle center. The following approaches are usually helpful:

1. For turning operations look down onto the top face of the cutting tool.
(For inverted tooling this involves looking up at the tool from below.)
2. For milling operations look along the machine spindle centerline toward the surface being machined.

It should be noted that this technique does not always correspond with the definition adopted by the control system's manufacturer. A simple trial program entered into the machine will clarify the situation if it cannot be determined from the machinery manuals.

Determining clockwise and counterclockwise direction on milling machines with multiple plane circular interpolation capability can be done using the following rule and Figure 8.26.

Clockwise and counterclockwise are determined by looking at the plane of the circle from the positive end of the coordinate axis normal to the plane of the circle.

The arc center definition for circular interpolation is also a standard requirement. It informs the control of the point of rotation for an arc along standard axes from the arc start point or program datum. Use the following definitions and Figure 8.27 to determine proper arc center addresses needed for various planes of rotation.

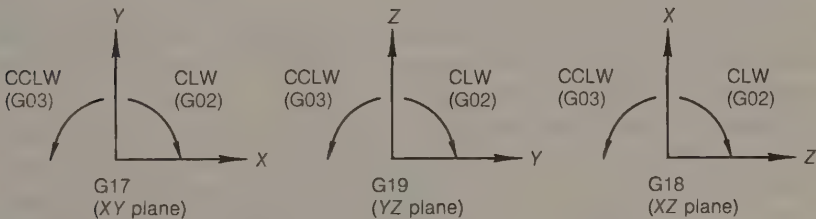


Figure 8.26 Determining clockwise and counterclockwise directions.

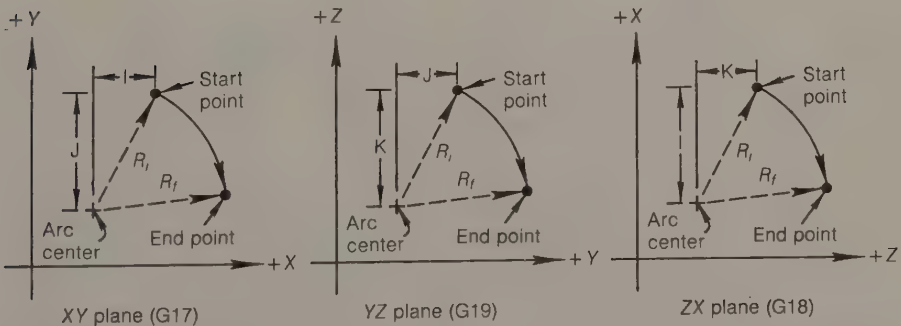


Figure 8.27 Arc centers.

Use of I, J, and K to Specify the Arc Center

I is the *increment* along the *X* axis from the start point of the arc to the arc center.

J is the *increment* along the *Y* axis from the start point of the arc to the arc center.

K is the *increment* along the *Z* axis from the start point of the arc to the arc center.

Note: Refer to your machinery manual to determine whether or not positive or negative signs will be required on the I, J, K values.

QUADRANT CIRCULAR INTERPOLATION

Quadrant circular interpolation allows up to 90° of arc to be programmed as long as the entire segment does not cross quadrant lines; refer to Figure 8.28

For any circle there are four quadrants that are created by axis lines which cross at the center of the circle. These axis lines are parallel to the part coordinate axes. No circular block may cross either of these axes. Starting at any point on the circle, it can be seen that the maximum number of circular controller data blocks to cut a 360° arc is five.

An individual circular controller data block cannot cross an axis line. To continue into the next quadrant, terminate the present circular controller data block and program another circular controller data block, starting on the axis.

This procedure is quite straightforward.

A minimum of four sets of dimensional data are required to complete a full circle with up to four dimension words per block. Two dimension words are required to define the arc and one or two are required to define the arc center. Arc center values of zero are normally not required. Machines that are capable

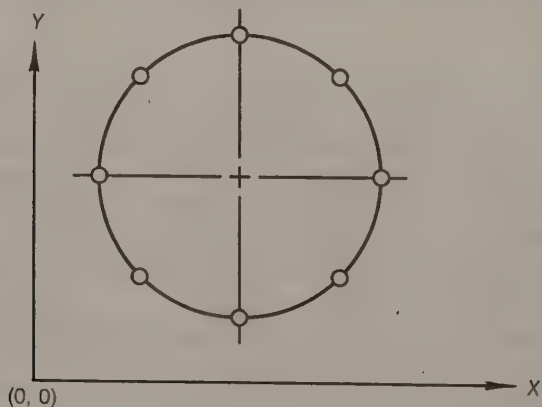


Figure 8.28 Circle quadrants.

of either quadrant or 360° circular interpolation will require a modal G code to be programmed (G74 quadrant; G75 360°). *Note:* For exact programming format and arc size limits refer to your machinery manual.

The two arc center address programming techniques previously referred to can be described using the component detail shown in Figure 8.29, and assuming that the last programmed move has brought the cutting tool to the start point indicated.

Method 1

1. The target or finish point of the arc is dimensionally defined, using *X*, *Y*, or *Z* values, in relation to the program datum when using absolute mode, or to the finish position from the previous move when using incremental mode.
2. The center of the arc is dimensionally defined in relation to the start point using *I*, *J*, and *K* values measured along the *X*, *Y*, and *Z* axes respectively.

Using this method the arc shown in Figure 8.29 would be programmed as follows:

Inch:

In absolute terms: G02 X1.6 Z1.6 I0 K.8 (Diameter programming of lathe part.)

In incremental terms: G02 X.8 Z-.8 I0 K.8

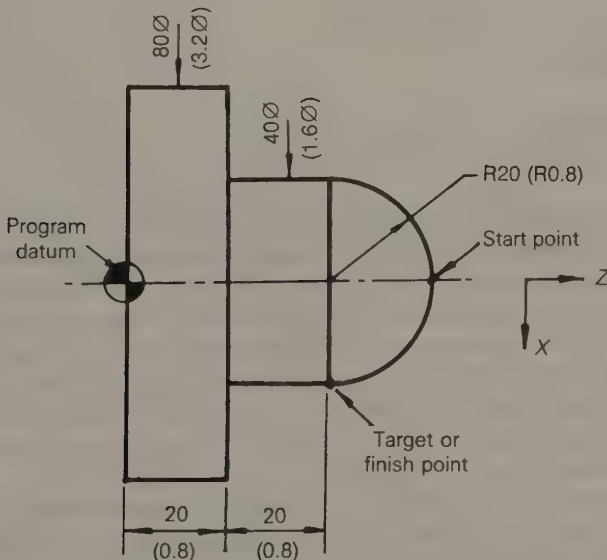


Figure 8.29 Component detail. (Inch units are given in parentheses.)

Metric:

In absolute terms: G02 X40 Z40 I0 K20 (Diameter programming of lathe part.)

In incremental terms: G02 X20 Z-20 I0 K20

Note that the I code has no value because the center and start point of the arc are in line with each other. In practice, when a value is zero, it is not entered into the program. In this method of arc definition the I, J, and K values are unsigned.

Method 2

The second method of word address arc programming differs in the way the arc center is defined. The following data are required:

1. The target or finish point of the arc is dimensionally defined, using X, Y, and Z values, in relation to the program datum when using absolute mode, or to the finish position from the previous move when using incremental mode.
2. The center of the arc is dimensionally defined in relation to the program datum using I, J, and K values measured along the corresponding X, Y, and Z axes, respectively, in absolute mode. The arc center is still defined from the previously defined point in incremental mode.

Using this second method of programming the arc shown in Figure 8.30 would be programmed as follows:

Inch:

In absolute terms: G02 X1 Y2.4 I1 J1.4

In incremental terms: G02 X1 Y1 I1 J0

Metric:

In absolute terms: G02 X25 Y60 I25 J35

In incremental terms: G02 X25 Y25 I25 J0

It is possible when using this approach for the I, J, and K values to be negative, as illustrated in Figure 8.31. These values are, therefore, signed plus or minus.

The two arc programming methods described will cater for movement within one quadrant only with each block of program data. Thus programming a complete circular move would require four blocks of data minimum. Similarly, blending arcs would require a separate block of data for each quadrant involved. This latter situation is illustrated in Figure 8.32; the first block would take the tool from point A to point B, and the second block would continue the movement from point B to point C.

When the start or stop points or both do not coincide with an X, Y, or Z axis, that is the arc is not exactly 90° or a multiple of 90°, it will be necessary to make a series of calculations.

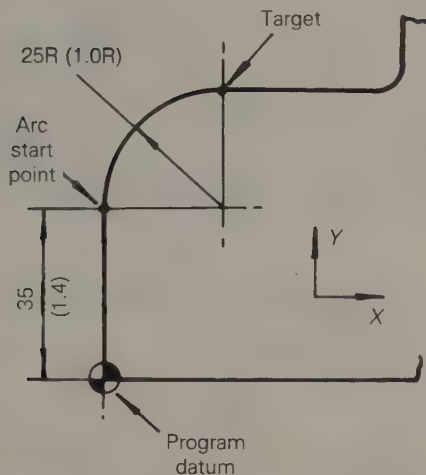


Figure 8.30 Milled component detail. (Inch units are given in parentheses.)

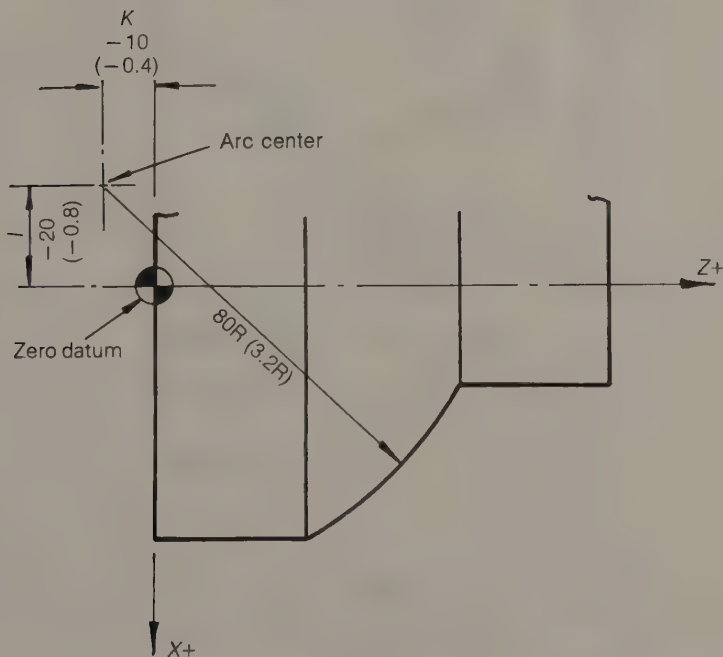


Figure 8.31 Circular interpolation: negative I and K values for an absolute program. (Inch units are given in parentheses.)

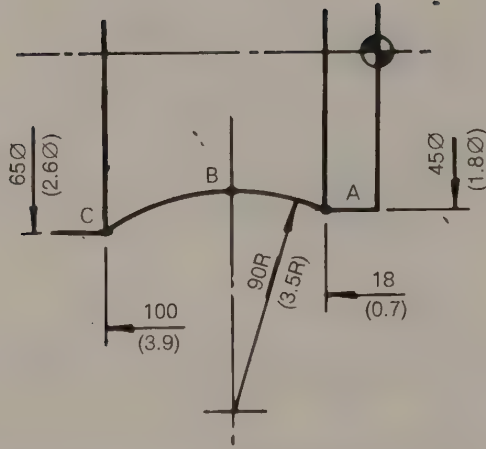


Figure 8.32 Profile detail requiring two arcs to be programmed. (Inch units are given in parentheses.)

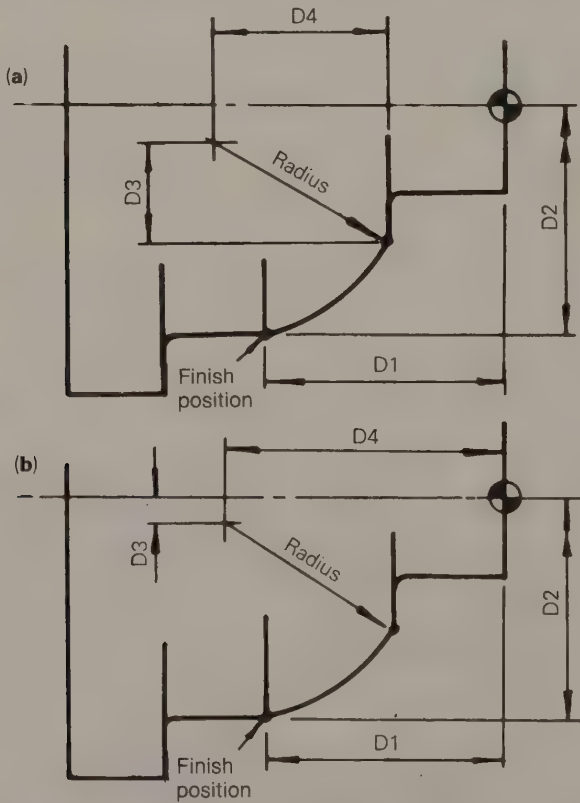


Figure 8.33 Calculations for arcs less than 90°: (a) method 1; (b) method 2. (Inch units are given in parentheses.)

Consider the part detail shown in Figure 8.33(a) and again in Figure 8.33(b), each diagram relating to an arc programming method as indicated. Whichever of the two methods is used, the finish arc position indicated by the dimensions D1 and D2 will have to be defined dimensionally. It will also be necessary to calculate the additional dimensions indicated on each drawing as D3 and D4, these latter dimensions being expressed in the part program as I and K values. A number of circular interpolation examples follow.

PROGRAMMING EXAMPLES

The circle in Figure 8.34 will be used to illustrate programming various arcs, clockwise and counterclockwise, absolute and incremental. (All units in the following examples are in inches.)

Example 1 (Method 1)

Go from P2 to P5 clockwise (absolute) (G74 default).

Signed I and J

```
G17
G90
G2X3Y2I-.7071J-.7071F10
X2Y1I-1
X1Y2J1
```

Unsigned I and J

```
G17
G90
G2X3Y2I.7071J.7071F10
X2Y1I1
X1Y2J1
```

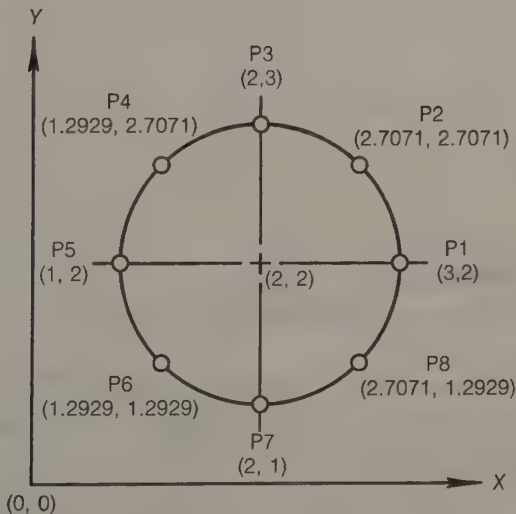


Figure 8.34 Example of programming arcs.

Example 2 (Figure 8.35)

Go from P2 to P5 counterclockwise (incremental) (G74 default).

Signed I and J	Unsigned I and J
G17	G17
G91	G91
G3X-.7071Y.2929I-.7071J-.7071F10	G3X-.7071Y.2929I.7071J.7071F10
X-1Y-1J-1	X-1Y-1J1

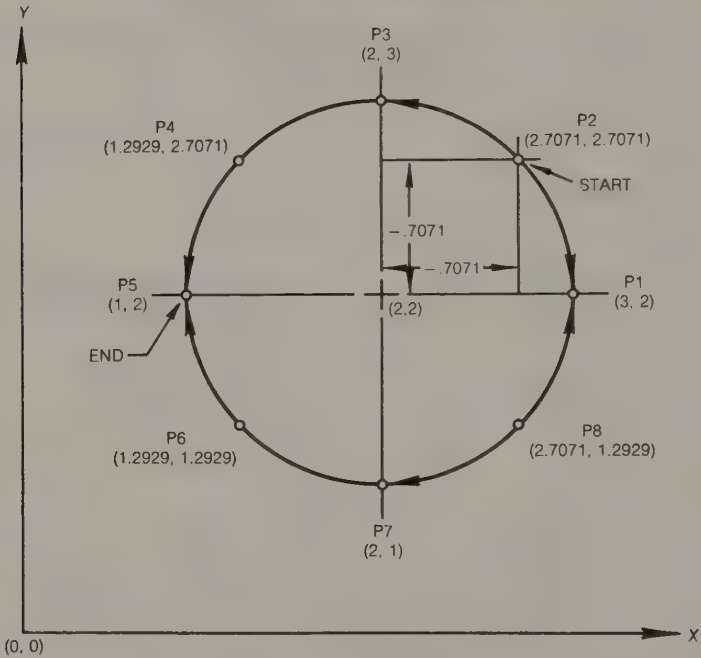


Figure 8.35 Examples 1 and 2.

Example 3 (Figure 8.36)

Go from P4 to P7 clockwise (incremental) (G74 default).

Signed I and J	Unsigned I and J
G17	G17
G91	G91
G2X.7071Y.2929I.7071J-.7071F10	G2X.7071Y.2929I.7071J.7071F10
X1Y-1J-1	X1Y-1J1
X-1Y-1I-1	X-1Y-1I1

Example 4 (Method 1) (Figure 8.36)

Go from P4 to P7 counterclockwise (absolute) (G74 default).

Signed I and J

G17
G90
G3X1Y2I1.7071J-.7071F10
X2Y1I1

Unsigned I and J

G17
G90
G3X1Y2I.7071J.7071F10
X2Y1I1

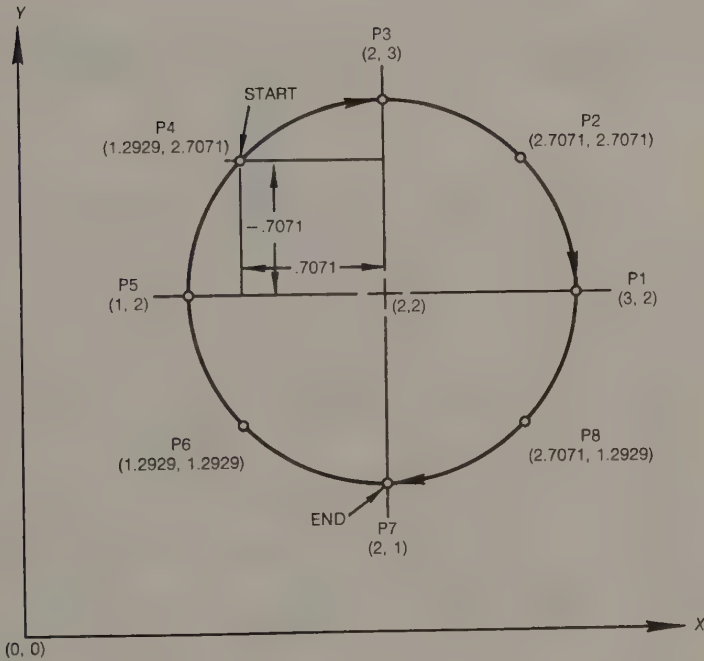


Figure 8.36 Examples 3 and 4.

Method 3

Single block or 360° circular interpolation is now available on more sophisticated CNC controls. Single block circular interpolation allows up to 360° of circular movement to be programmed in a single block of information when the modal G75 code is active. All previous circular interpolation command words defined are still used to determine rotation, finish/end position, and arc

center. It should be noted, though, that in this form of circular interpolation the I, J, and K combination used must be signed to determine the circle's center.

Another difference that will be noted in this method of programming is if a full 360° of circle is to be produced, the starting and ending points of rotation will be identical. When identical starting and ending points occur, the end point of the arc does not have to be programmed, only the rotation and arc center information are required; refer to the examples with Figure 8.37.

Example 5 (Method 3) (Figure 8.37)

Go from P3 to P3 clockwise.

Incremental	Absolute
G17	G17
G91	G90
G75	G75
G210J-1F10	G0210J-1F10

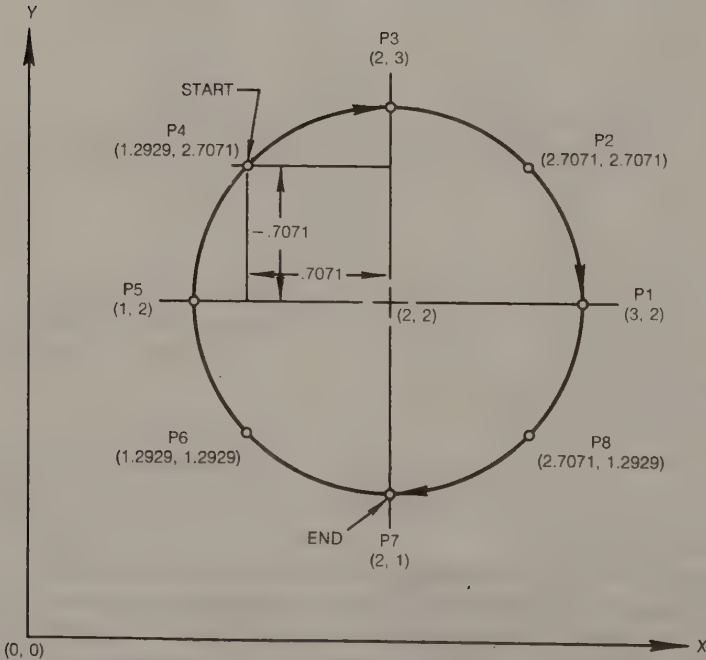


Figure 8.37 Examples 5 and 6

Example 6 (Method 3) (Figure 8.37)

Go from P4 to P7 clockwise.

Incremental	Absolute
G17	G17
G91	G90
G75	G75
G2X.7071Y-1.7071I.7071J-.7071F10	G2X2Y1I.7071J-.7071F10

TOLERANCED DIMENSIONS

It is often the case that dimensions on drawings are toleranced, thus permitting a higher and lower limit. Since it is only possible to enter one value into the control unit, it is logical that this should be the middle or mean value of the tolerance band.

REPETITIVE MACHINING SEQUENCES

There are a number of repetitive sequences that are commonly used when machining a variety of components like drilling a hole. Other less common sequences are also repetitive, but on only one particular component (hole patterns—milling path). It is helpful, since it reduces the program length and also simplifies programming, if such a sequence can be programmed just once; it is then given an identity so that it can be recalled into the program as and when required.

Repetitive machining sequences can be generally classified as follows:

- (a) Canned or fixed cycles that are a built in feature of the machine control system.
- (b) User or programmer defined routines to suit the particular job in hand (custom canned cycles, program macros).

The facility for the programmer to devise special routines may be restricted, especially on small training machines. However, even the most simple system will usually include one or two canned cycles. The controls fitted to advanced machines will have many available.

CANNED CYCLES

In a text of this nature it would be impossible to deal in detail with all the canned cycles that are available, but the review that follows will convey a good impression of the range currently in use and likely to be encountered. More

specifically, the student is advised to make a careful study of the programming methods and techniques associated with the control system he or she will be using. In this respect a close examination of the examples found in the programming manuals will be found to be helpful. The point to remember is that the use of canned cycles is an aid to programming efficiency and accuracy, and they should be used whenever possible. You will now notice that the previously mentioned special cycles for conversational programming are not necessarily unique to that form of programming. Word address programming has many of the same options that are used through the use of special "G" codes or possibly a switch on the control rather than a push button.

Fixed or canned G code cycles (G81–G89) were developed, as stated, to simplify the programming of hole-making operations with repetitive motions at each hole location. These codes are normally modal and must be cancelled with a G80 code (linear rapid transverse mode) before other program movements are commanded. The normal action of the canned cycles mode is to position the programmed nonspindle (tool penetration) axes, including the rotary table axes on milling machines, with the (Z) coordinate axis remaining stationary. The tool will then rapid the Z axis to the R or tool-clearance plane preceding the part surface and feed at the programmed rate to Z depth. After feeding the tool to depth, the tool will be retracted from the hole automatically as required by the cycle. Note: different codes will cause various forms of retraction (refer to your programming manual for specific retraction action). Once all of the necessary modal information is programmed (G code, depth, clearance position, spindle speed, feed, and spindle direction), additional holes can be produced by programming new axis or axes positions until the canned cycle is cancelled. Most canned cycles will pick up the last programmed spindle speed/direction and feedrate if they are not initiated in the canned cycle block of information.

Perhaps the most widely used machining sequence is that of drilling a hole, and there are few controls that fail to cater to this requirement by including a canned cycle. Indeed, with word address programming, early attempts were made to standardize a drill cycle. That this was quite successful is evident by the fact that the use of G81 for the purpose is as common as the use of G00 and G01 for linear movement control and G02 and G03 for circular interpolation.

There are a number of machining variations necessary in the production of drilled holes.

One of the most commonly used is the basic drilling movement, catered for by the drilling cycle illustrated in Figure 8.38. This involves a drill movement to the required depth at a controlled feed rate, followed by rapid withdrawal.

Also widely used is the intermittent or "peck" drill cycle for deep holes illustrated in Figure 8.39. This illustration shows a complete withdrawal to the Z axis clearance plane after each peck, but variations of this cycle provide for a smaller withdrawal that conveniently breaks the chip but does not give total

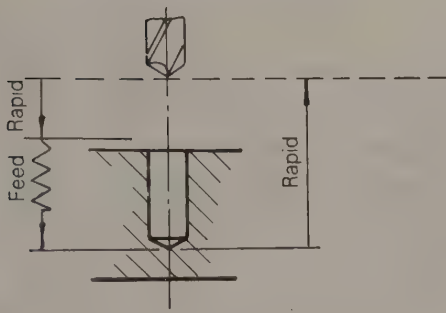


Figure 8.38 *Drill cycle.*

retraction for chip clearance. The peck depth is established with an additional Z axis command word, which many times is a “K” value.

A further refinement of this cycle provides for automatic variation of the peck length as the hole deepens. This is achieved by including a “multiplier” in the cycle data. For example, a multiplier of 0.8 will have the effect of reducing each peck length to 0.8 of the previous peck. To avoid the reduction continuing indefinitely a minimum peck length is also programmed.

Closely allied to the drilling cycles are those for boring and tapping. To ensure a clean surface, boring may require special or no drag line retraction, and counterboring may require the inclusion of a time dwell at the extent of cutter travel. Tapping requires that the direction of spindle rotation is reversed to allow withdrawal of the tap.

Many turning operations involve machining chamfer or radius features illustrated in Figures 8.40(a) and 8.40(b). It is possible to program a special cycle on some machines that would normally require two data blocks in just one block. Figure 8.40(a) shows the programming case which would otherwise require a Z axis movement followed by simultaneous movement in both the X and Z axes to produce an angle or chamfer. Figure 8.40(b) shows a similar

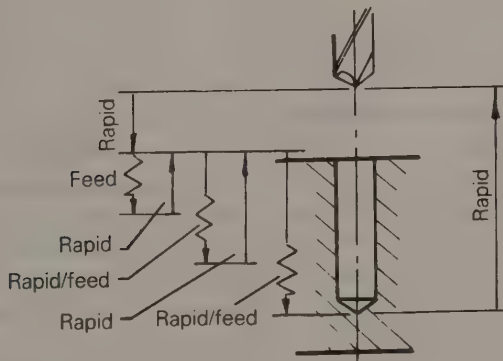


Figure 8.39 *Peck drill cycle.*

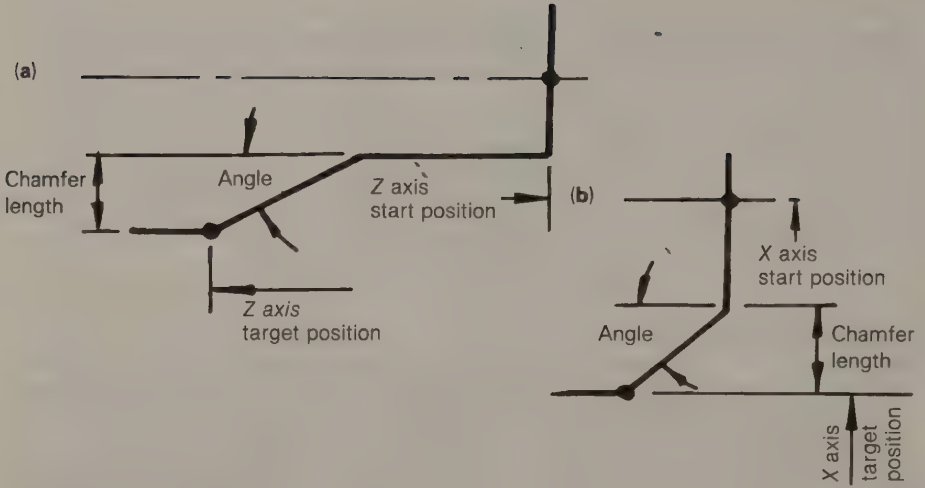


Figure 8.40 Chamfering cycles.

situation, but in this case the angular movement is preceded by linear movement in the X axis.

Very similar to the cycles described previously are those illustrated in Figures 8.41(a) and 8.41(b). Instead of a linear move being followed by an angular move, the linear moves are followed by radial movement. In both cases the radial movement must be a full 90°.

Both the chamfering and radius cycles may be automatic with little or no programming required for a standard size or there may be special G code and axis information required. Refer to your specific programming manual.

Automatic threading cycles are the counterpart of the G84 tapping canned cycle for single-point threading tools. Single-point thread cutting tools require

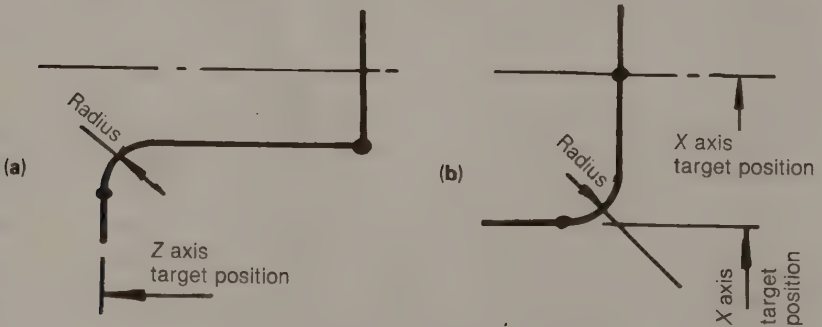


Figure 8.41 90° arc cycles.

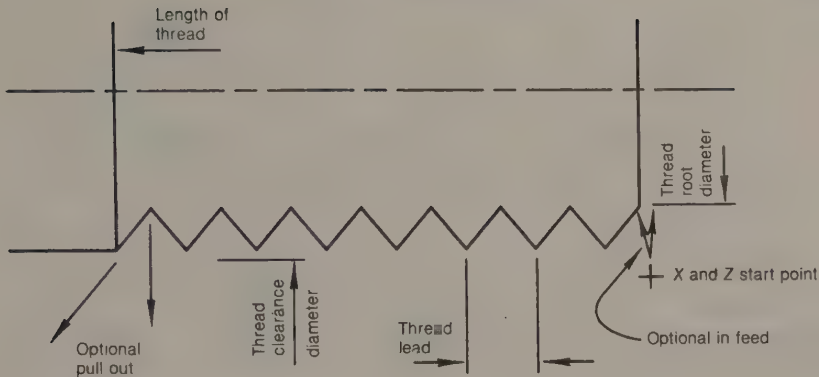


Figure 8.42 Automatic threading cycle.

a number of passes to produce a finished thread. Special G codes, like G33, have been developed for lathe work, which allows the programmer to call thread approach and depth, cut distance, retract to clear part, and return to start point all in one block/line of information. Various controls will handle this differently, some requiring a line of instruction for each thread pass and others allowing the entire thread cutting operation to be cut with one statement. G codes like G28 and G29 may be used to define the basic thread pass for later recall. The various codes have been developed to handle both internal and external threads as well as straight or tapered threads. Figure 8.42 shows an example of the types of program information required to make this cycle work.

USER-DEFINED ROUTINES

Canned cycles cater for the easy programming of machined features that are often required on a wide range of components. But the part programmer is often confronted with a feature that is repeated a number of times on a particular component but is found only on that component or a limited range of components. It is in situations such as this that the facility to devise a special routine for use as and when required is very helpful.

Consider the component detail shown in Figure 8.43. Along the length of the shaft is a series of identical recesses. If there is no facility to write a special program, or routine, to machine these recesses the programmer is faced with the rather cumbersome task of detailing each move necessary to machine one recess and then repeating the data for each of the others.

When preparing a routine to accomplish a specific machining task such as this, the programmer can include any of the available canned cycles that might be appropriate. For example, the profile of the shaft recess referred to could be machined using the cycle that permitted one-block programming of a linear

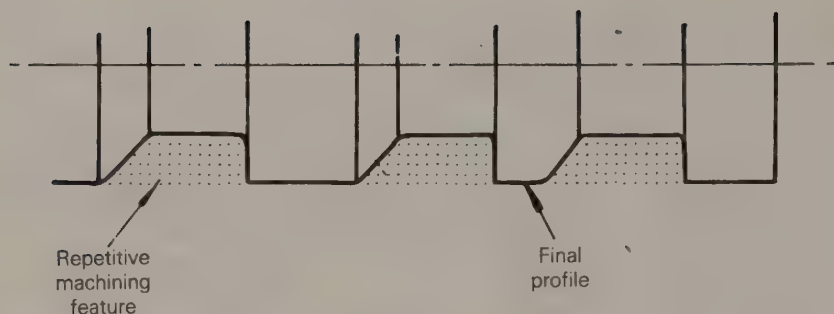


Figure 8.43 Application of a turning subroutine.

move in the Z axis followed by an automatic chamfer or radius. When specific routines are programmed within other routines, they are said to be “nested.”

Assume the component shown in Figure 8.44 has a repetitive feature as indicated that justifies using a specially devised routine to clear the recess and mill it to the required profile. Now assume that within each of these recesses there is a series of three smaller recesses as shown in Figure 8.45. Since there will be three times as many smaller recesses as larger recesses, a further specially devised routine to machine them will be justified.

The routine for machining the larger recess will therefore contain the subroutine for machining the smaller recess. The subroutine is nested within the first routine and will be activated three times during the machining of the larger recess.

It would be quite feasible for each of the smaller recesses to include a drilled hole, and this could also be produced using a canned cycle. The subroutine for the smaller recess would now include a nested drilling cycle.

There are usually some limitations regarding nesting. Some controllers permit subroutines within subroutines up to eight deep; others may accept only half this number.

Specially devised routines can also be used to control machine movements and functions not directly associated with metal cutting. For example, a special

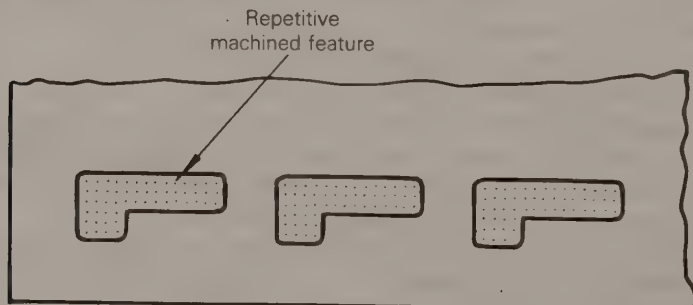


Figure 8.44 Component detail: application of a milling subroutine.

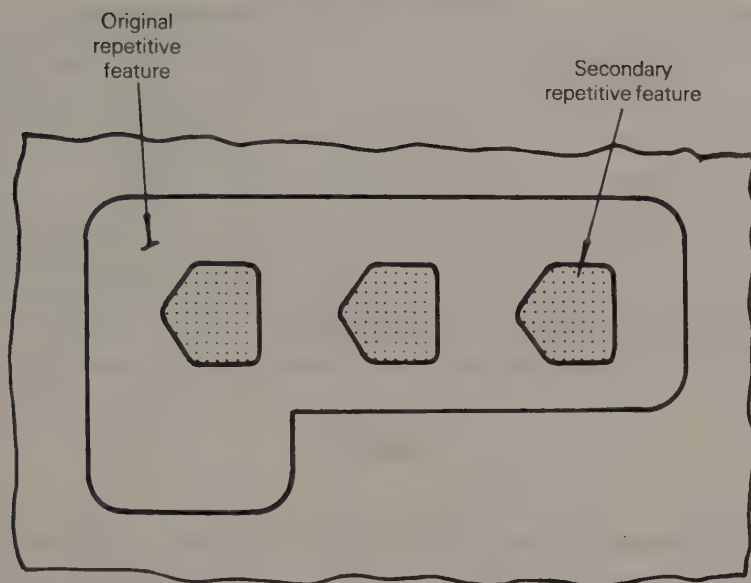


Figure 8.45 Nesting of subroutines.

routine can be used to establish and readily recall predetermined parameters relating to machine slide positions and programming modes which may, for safety reasons, need to be established from time to time throughout the program run. The application of a programmer-devised safety routine is included in the sample program listed in Figure 8.46.

The safety routine in this instance is determined by the blocks N10 to N70. It can be seen that, in the program, these data blocks are activated by a G25 program entry. Blocks N100 and N160 are examples where the safety routine is being called. Each time the routine appears in the program the slides return to a safe indexing position and a set of known operating modes is reestablished. This provides a basis from which the programmer can proceed with the programming of further machining operations.

Machine: Hardinge HXL Turning Center Control: GE 1050 Sample Program for Drawing Figure 8.46

N0010	G71	} SAFETY ROUTINE	Metric
N0020	G40		Cancels Tool Nose Radius compensation (TNRC)
N0030	G95		Feed mm/rev
N0040	G97 S1000 M03		Spindle rev/min. CW
N0050	G00		Cancels G01, G02, G03, etc.
N0060	G53X177.8Z254 T00		Return to safe indexing position, offsets cancelled
N0070	M01		Optional stop

N0100	G25P ₁ 10P ₂ 60	}	CENTER DRILL	Calls safety routine
N0110	T1200			Calls tool
N0120	G54X0Z3T1212			Move in X and Y with zero shift to work face and tool offset active
N0130	S2500F.1	}	CENTER DRILL	Spindle speed and feed
N0140	G01Z-6			Center drill
N0150	G00Z2			Drill retract
N0160	G25P ₁ 10P ₂ 70			Safety routine call with optional stop turret returns to safe indexing position

Note: In the programming areas of special canned cycles and routines, controls vary drastically and your programming manual should be reviewed.

LOOPS

Some control systems provide a “loop” facility. This enables the programmer to devise a routine and to repeat that routine within the part program a specific

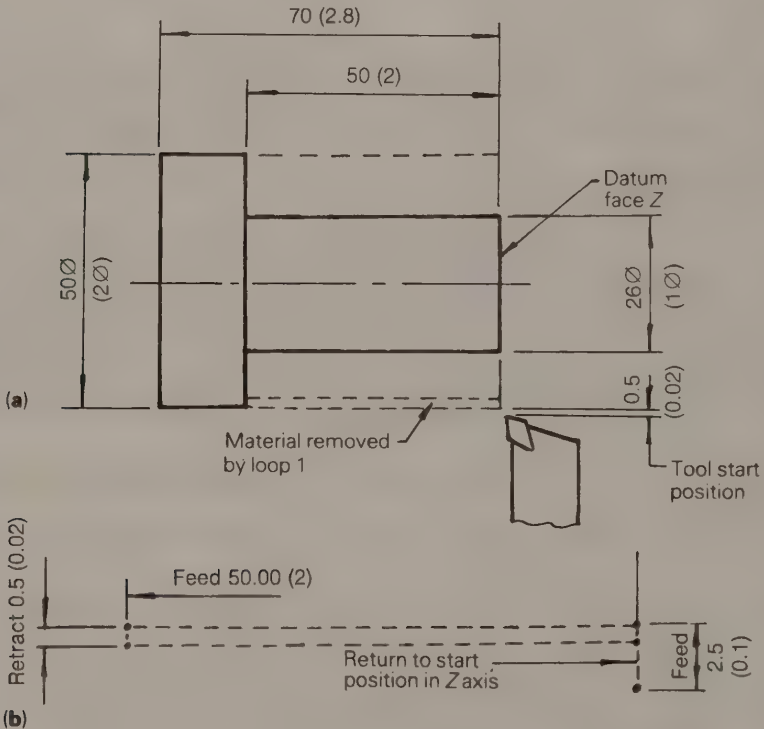


Figure 8.46 Looping cycle: (a) component detail, (b) loop detail, repeated six times. (Inch units are given in parentheses.)

number of times. In other words, when the program reaches the end of the routine the control will return, or loop, back to the beginning of the routine again.

Consider the component shown in Figure 8.46, which is to be reduced from 50 mm (2 in.) diameter to 26 mm (1 in.) diameter by a series of cuts each of 2 mm (0.08 in.) depth. Assuming the starting point for the tool is as shown, the tool will first move in to a depth of 2.5 mm (0.1 in.), thus taking a 2 mm (0.08 in.) depth of cut, travel along a length of 50 mm (2 in.), retract 0.5 mm (0.02 in.) and return to the Z datum, so completing a loop. It will then move in a distance of 2.5 mm (0.1 in.), feed along 50 mm (2 in.), retract 0.5 mm (0.02 in.) and return to the Z datum, and so on. The loop, including the feed rate, is programmed just once, but is repeated via the loop count data included in the main program as many times as necessary to reduce the work to the required diameter.

MACROS

A somewhat specialized type of programmer-devised routine is referred to as a “macro.” This facility can be used for machining a complete component or a feature of a component that, while not standard in the wider sense, is nevertheless frequently required. For instance such a feature may commonly occur within the production schedule of a particular company. The macro is given an identity and stored within a separate macro file, or memory, and is called into use as and when required, possibly as an element within a much larger machining program.

A macro may have fixed dimensions, or it may have parametric variables which enable the dimensions to be varied to produce different versions of a basic component. This technique is referred to as “parametric programming.” The parametric or variable version of macro programming is very useful when programming for a family of parts that have the same shape but vary in size.

PARAMETRIC PROGRAMMING

A parameter is a quantity that is constant in one particular case but variable in others. A simple engineering example of a parameter is the length of a bolt. One version of the bolt will have a certain length; all other versions will be identical, that is, they will have the same thread form, diameter, and hexagon head, but they will all vary in length. Thus, the length of the bolt is a parameter, constant in one particular case but variable in others.

Parametric programming involves defining parameters and then using those parameters as the basis for one part program that may be used to machine not only the original component but a number of variations as well.

Figure 8.47(a) shows a component the dimensional features of which have been defined as parameters using the symbol # and a number: #1, #2, #3, and so on.

Figures 8.47(b)–8.47(g) show six variations of the component, the variations being indicated. A range of components such as this is referred to as a family of parts.

The machine movements necessary to machine each of the variations are all included in the original component. Some components require exactly the same movements, but with varying lengths of travel. Other components do not require all of the movements to be made. Using the more usual programming techniques, the production of each component would require a separate part program. Using the parametric part programming technique, instead of defining each dimensional movement individually in the *X* and *Z* axes, the parametric reference is programmed. Thus, to turn along the stepped diameter, the entry in the main program, referred to as the “macro,” would read as follows:

```
N07 G01 X#4
N08 Z#2
```

These entries would suffice for all components requiring a stepped diameter. Equally, one entry using parametric identification would suffice for facing all the components to length or drilling the hole.

Having programmed all movements and the sequence in which they are to occur, it remains to define them dimensionally. The dimensional details are entered as a list at the start of the part program. Thus the parameters and their metric dimensional values for the original components would read as follows:

```
#1 = -50.00
#2 = -30.00
#3 = 30.00
#4 = 22.00
#5 = 10.00
```

As each parameter is called in the macro body, the programmed dimensional entry made previously will be invoked.

To machine any of the variations in the family of parts requires a simple amendment of the original parametric values. The parameters to machine the component shown in Figure 8.47(b) would be

```
#1 = -50.00
#2 = -40.00 (amended)
#3 = 30.00
#4 = 22.00
#5 = 10.00
```

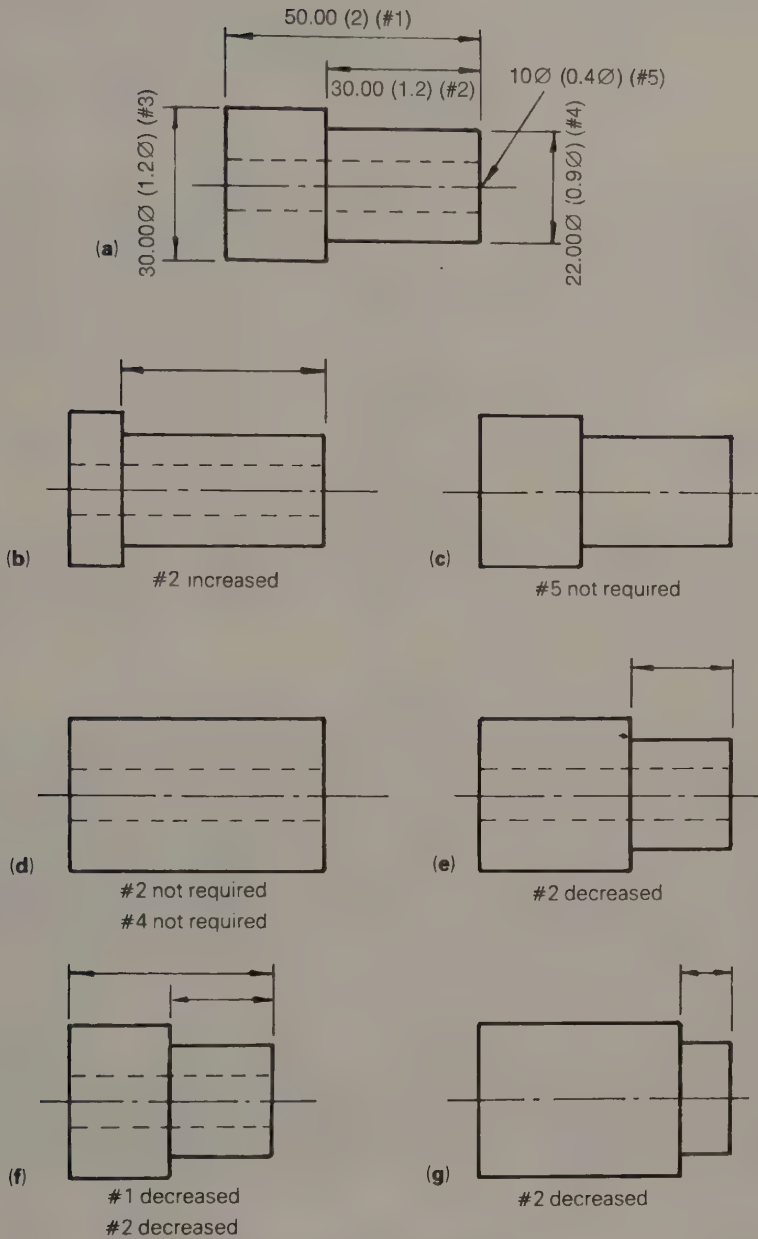


Figure 8.47 Parametric programming: a family of parts. (Inch units are given in parentheses.)

and to machine the component in Figure 8.47(f)

```
#1 = -40.00 (amended)
#2 = -20.00 (amended)
#3 = 30.00
#4 = 22.00
#5 = 10.00
```

Now consider the components where the programmed movements necessary for machining the basic component are not required. By using a relatively simple programming technique, the control unit can be caused to skip the redundant blocks. The necessary program entry involves the use of certain conditional expressions in which assigned abbreviations are used, such as the following:

```
EQ = equal to
NE = not equal to
GT = greater than
LT = less than
GE = greater than or equal to
LE = less than or equal to
```

Consider Figure 8.47(d) and assume the #1 and #3 have been machined. In the macro body the next call will be to machine the stepped diameter. To avoid this, blocks must be skipped and so an entry in the macro body will read as follows:

```
N15 IF [#4 EQ 0] GO TO N18
```

This statement says that if #4 is zero, move on to block number 18. Since #4 is nonexistent in the component, the parametric value will be entered as zero and, consequently, the control unit will move ahead.

The preceding description of the use of the parametric programming technique is a very simple one. It is in fact a very powerful concept and its full application is quite complex. For instance, parameters may be mathematically related within the macro body, that is, they may be added together, subtracted from one another, and so on.

In addition, the parametric principle may be extended to include speeds and feeds, when all the likely variations for roughing, finishing, etc., may be given a parametric identity and called into the program as and when required. *Note:* parametric and macro programming, if available on particular machinery, very drastically in programming techniques, so consulting the programming manual is required.

POINT DEFINITION

Point definition is a programming facility, not widely available, which simplifies programming for drilling operations. With this facility it is possible to

define dimensionally as many as 99 points or positions, and enter them into a special file within the control memory. The file can be accessed as required, the points' positions appearing in tabular form.

The points required for inclusion are entered at the start of a part program, and might look as follows:

N1	G78	P1	X15	Y20
N2	G78	P2	X20	Y20
N3	G78	P3	X50	Y30
N4	G78	P4	X65	Y60
N5	G78	P5	X75	X75
N6	G78	P6	X98	Y78

To drill a hole to a specified depth of 20 mm, using a G81 drilling cycle at points 2, 5, and 6, would require a program entry as follows:

N095	G81	Z-20	F150	S1850
N100	G79	P2	P5	P6

The more holes to be drilled, the more advantageous the use of the facility becomes. The dimensional data relating to each point can be modified to suit any particular job.

MIRROR IMAGE

Mirror image is the term that describes a programming facility used to machine components, or features of components, that dimensionally are identical but geometrically opposite either in two axes or one axis. By using the mirror image facility such components can be machined from just one set of data.

In Figure 8.48 an original component feature is indicated in the bottom left-hand corner of the diagram. A complete mirror image of that feature is shown in the top right-hand corner, while mirror images in the *X* axis and the *Y* axis are shown, respectively, in the bottom right- and top left-hand corners of the diagram.

To produce a complete mirror image both the *X* axis and *Y* axis dimensional values will change from negative to positive. For half mirror images the dimensional values will change from negative to positive in one axis only. Mirror imaging is normally achieved by repeating a series of program instructions after activating a *X* and/or *Y* axis switch or button on the machine controller.

TOOL OFFSETS

Most machining operations involve the use of more than one tool, and usually they vary in length and/or diameter. To accommodate these size variations,

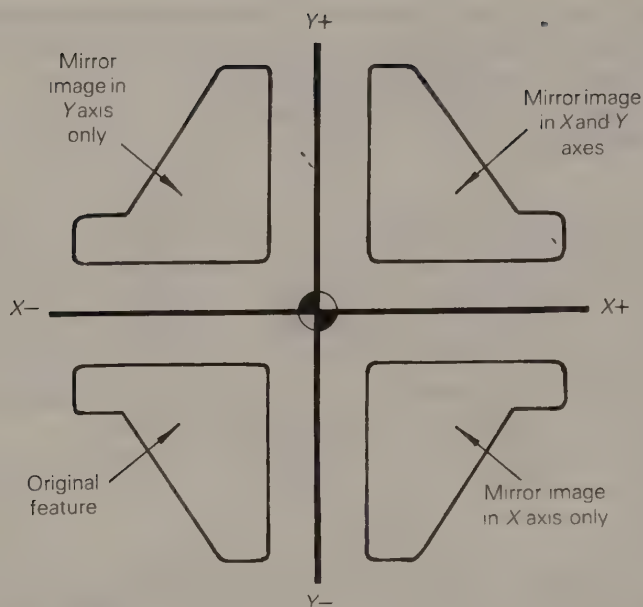


Figure 8.48 *Mirror image.*

and to permit the programmer to assume that all the tools are identical, machine controls are provided with a cutter compensation facility that will, when activated, automatically adjust the programmed slide movements. Thus, it enables the programmer to totally ignore tool size and simply program movements that are exactly the same as the profile detail, a technique referred to as “point” programming or zero-gage-length programming. Some programmers do not agree with this form of programming owing to the inherent dangers if a compensation is missed or gage length incorrectly measured. Therefore, in gage length programming techniques where tool length and radius/diameter are allowed for in program calculations, the tool compensation is used for variations in setup dimensions versus programmed tool lengths.

The task of dealing with variations in tool size is left to the tool setter or operator, since it is essentially a case of ascertaining the tool sizes or size variations and entering these numerical values into the machine control.

The numerical values that are required to be entered relate to tool length and tool radius/diameter.

The manner in which tool length variations are determined and entered varies with machine type. On some controls they are entered simply as offset values, one tool being used as a reference tool and thus having no offset value, and all other tools having data entries corresponding to their dimensional variation from the reference tool. This principle is illustrated in Figure 8.49.

On other machines the size variations of all tools are determined in relation to a fixed point on the machine, such as the corner of a tool post, as illustrated

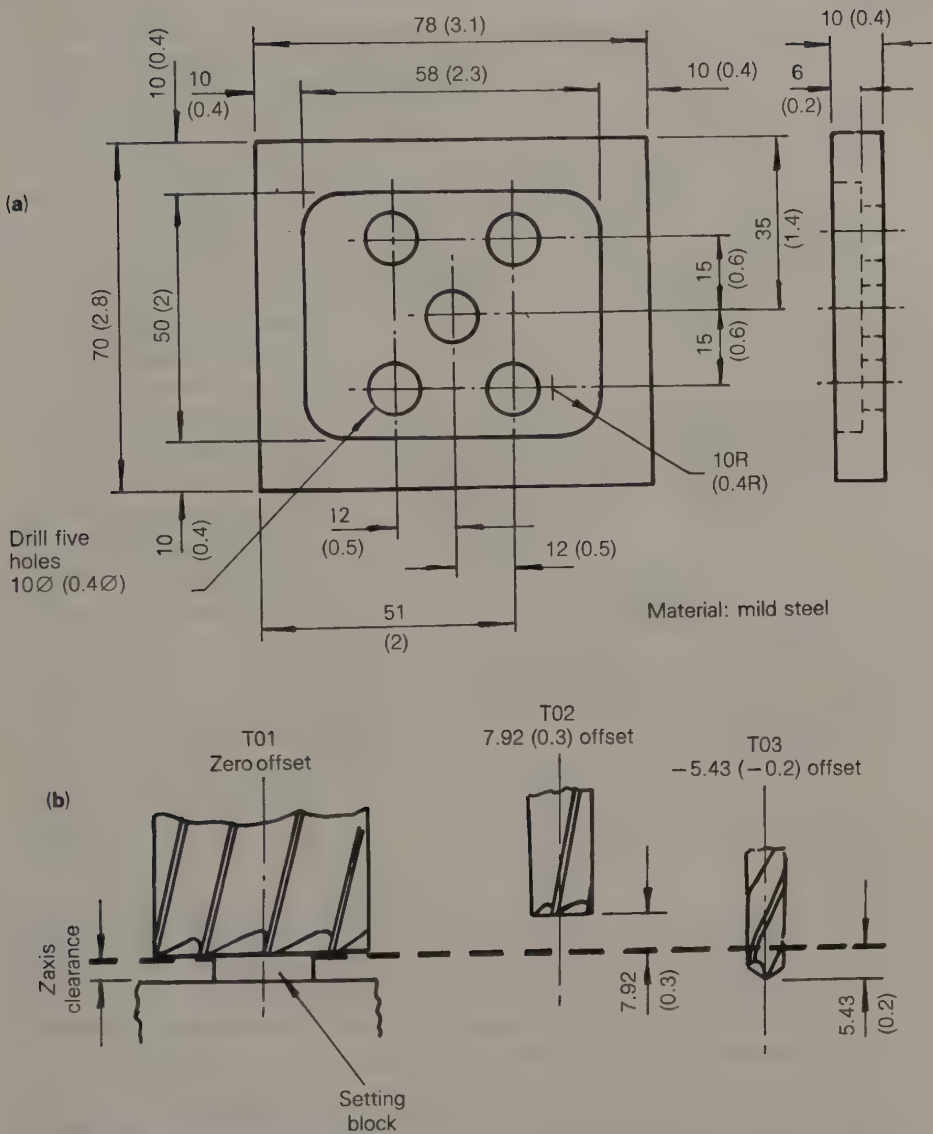


Figure 8.49 Tool offsets related to a reference tool: (a) component detail; (b) tool offsets. (Inch units are given in parentheses.)

in Figure 8.50. In this particular example the variations are entered in a tool data file, there being a second file for programmer-devised offsets that are, through appropriate program data, paired with the tool data file entries.

Tool radius and diameter entries are less complex, being simply a case of ascertaining the correct value, by actually measuring the tool if need be, and

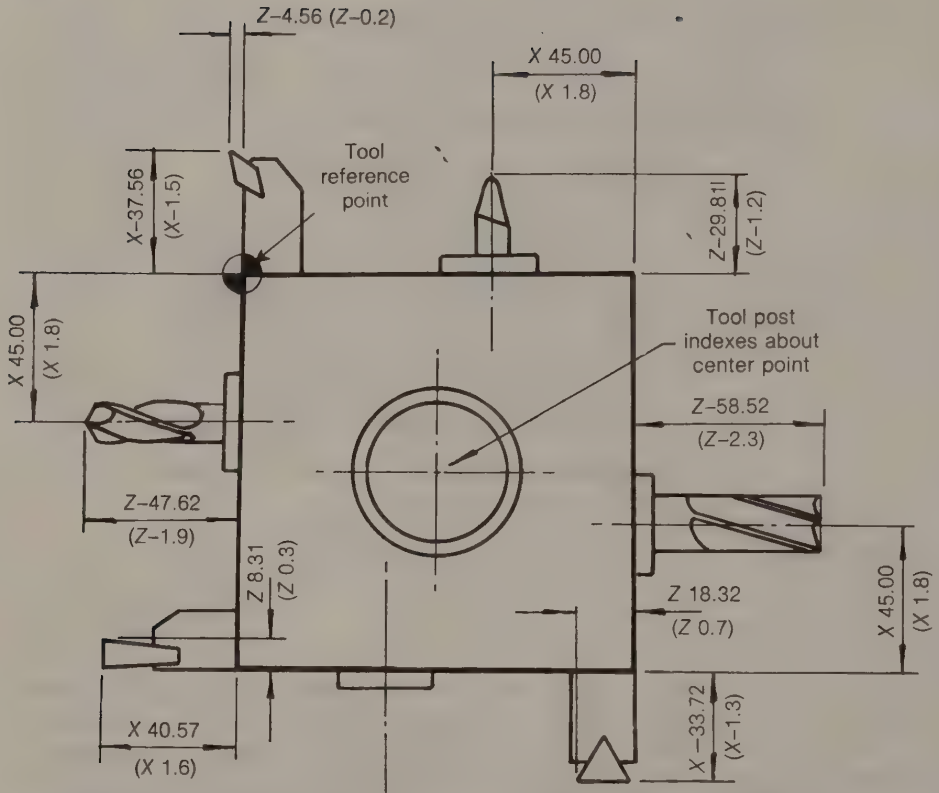


Figure 8.50 Tool offset data related to a fixed point on the toolpost. (Inch units are given in parentheses.)

entering the numerical value as required by the control system of the machine being used.

Since tool data can be entered, modified, or erased by the machine operator at will the facility can be used to

- accommodate replacement tooling that varies from the original;
- make variations to the component size;
- initiate a series of cuts, say roughing and finishing, using the same dimensional programmed data.

If the machine operator varies the tool offset values for whatever reason, the effect is temporary. It has no permanent effect on the original program. However, the programmer can utilize the offset facility within the part program, and this creates a situation where accurate communication between the programmer and the shop floor personnel is essential if the programming objectives are to be clearly understood.

Two situations involving tool offsets that are particularly useful from a programming point of view are when variations in the sizes of components are to be made from the same basic program and when a series of cuts are to be initiated along a profile using the same programmed dimensional data.

The ability to use offsets in this manner is based on pairing offset values with specific tools. Just as tools are allocated a numerical identity, so are offsets. Two digits are commonly used, just as in the case of tool identity. The offset digits are paired with the tool digits and are included in the program as part of the tool call. Thus, tool number three with offset number six would be entered in the program as T0306. Older word address programming systems may have special letter codes for calling the tool offset active, such as "H" for length and "D" for diameter.

Control systems usually provide for more offset entry capacity than there will be tools available, so it is possible to call any offset with any tool. The ability of programming any offset with any tool also allows programming more than one offset per tool. An example when to use this feature is when one tool is used to create more than one critical dimension.

The technique of using a number of offsets to make a series of cuts along a profile is illustrated in Figure 8.51. (A similar technique can be used when milling profiles and this is discussed later in the text.)

While the use of offsets as described previously is a very useful programming facility, it should be remembered that the prime objective of an offset facility is to make zero-gage-length programming a possibility and to allow for variation in cutter size during replacement simplifying the programming process. The preceding text has dealt only with tool lengths. It is now necessary to consider the way in which a variety of cutter diameters and tip radii can be accommodated.

To facilitate zero diameter programming with a variety of cutters of varying radii, the control should move the cutter away from the work profile a distance equal to its radius. This facility is referred to as "cutter radius compensation" or "cutter diameter compensation." This compensation also allows for variation in size do to cutter sharpening or deflection.

The distance the cutter will actually move away from or toward the profile—the offset—will be related to the data entry made by the machine setup person or operator. The offset is activated via the appropriate program entry.

The offset can be programmed to occur to the right or left of the required profile, commonly by the use of G41 or G42 when programming in word address mode. To determine which offset code should be programmed, the technique is to imagine being the tool and facing the direction of tool travel. The tool can then be visualized as being either to the right or left of the profile. It is very important to ensure that the correct offset is programmed since a move in the wrong direction may have disastrous results, particularly when large diameter cutters are being used. Tool radius to the right and left of a profile is shown in Figure 8.52.

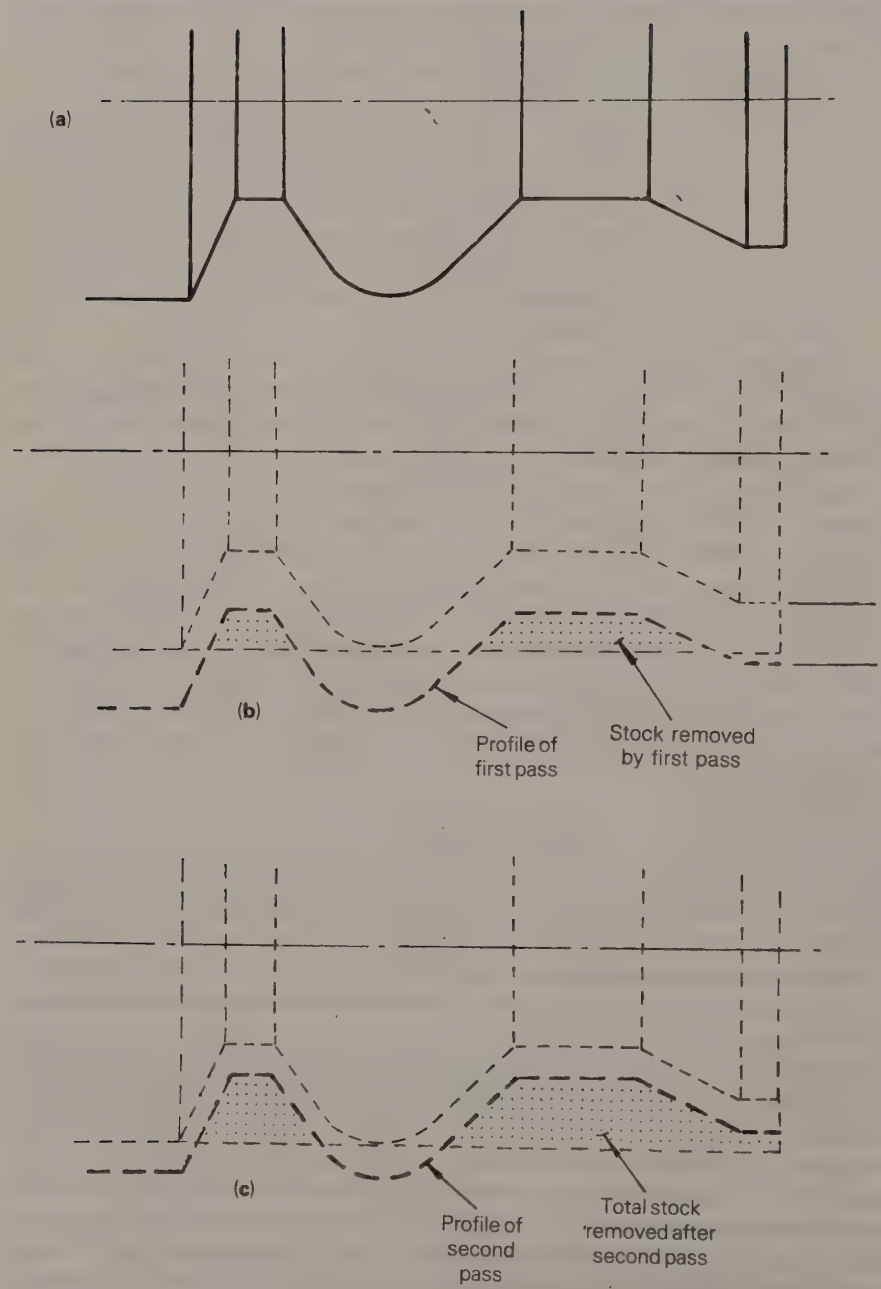


Figure 8.51 Use of tool offsets for progressive stock removal to a set profile: (a) profile detail; (b) first cut; (c) second cut.

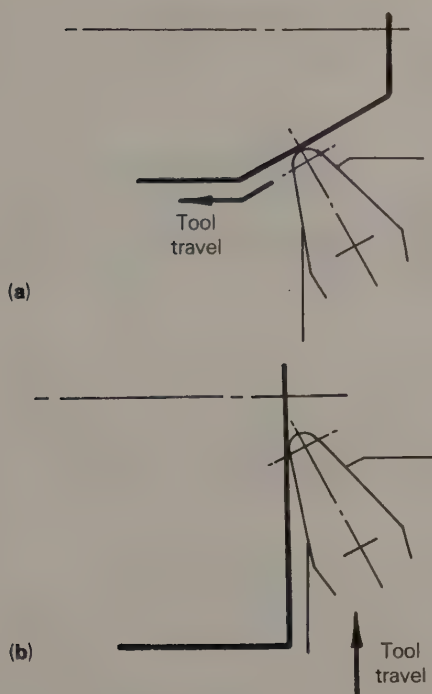


Figure 8.52 Tool nose radius compensation (TNRC): (a) radius compensation left; (b) radius compensation right.

When activating cutter radius compensation, it must be ensured that the slides will first make a noncutting move, to enable the correct tool and workpiece relationship to be established. A similar move is necessary prior to cancellation of the radius compensation. These noncutting moves are referred to as “ramp on” and “ramp off,” respectively.

It is now possible to return to the technique referred to earlier, of using offsets to make a series of passes along a milled profile. It is achieved by simply entering a bogus value for the cutter diameter into the control system. By making an entry that is greater than the size of the cutter being used, the actual offset activated via the program will be greater. Thus, the final profile will remain oversize as illustrated in Figure 8.53. The technique can also be used progressively to remove surplus material before making a final cut.

The reverse application of this technique, that is, entering a value smaller than the actual cutter size, will result in a smaller offset and, in the case illustrated, an undersize component profile. Thus, it is possible to produce components of varying dimensions from the same program when milling, just as it is possible to do so when turning.

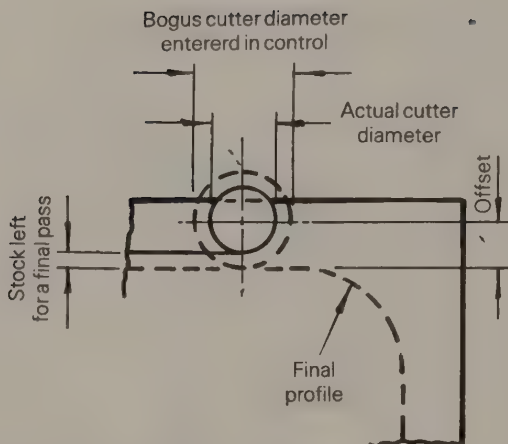


Figure 8.53 Using an offset to create an oversize milled profile.

BLOCK DELETE

Production engineering often involves machining a range of components that have slight variations from each other. For example, a hole that has to be drilled in one component is not required in a second component, although all the other details remain the same. Thus, one program would serve for both components providing that some means exists for not drilling the hole when it is not required. The way this is achieved is by use of a “block delete” facility.

Blocks relating to machining features that may not always be required incorporate the symbol /, which is referred to as a slash. The exact position of the slash within the block may vary from one system to another, but it is usually at the start before the sequence number (/N0245).

The machine operator will need to be instructed as to whether the data are to be retained or deleted from the current machining task. If the data are to be retained, the operator takes no action. If the data are to be deleted, the operator has to activate the block delete button on the control panel before running the program. Activation of the button is usually indicated by a light.

If the slash delete button is not activated, the control will respond to all the data contained in the program. If the slash delete button is activated, then all the blocks containing a slash will be ignored.

On some control systems if the slash delete button is not activated, the program will automatically stop when the first slash is reached and the operator then has to make a positive response, either to activate the data contained within the slashes or delete them.

The block delete facility is also useful when machining castings or forgings, where stock removal requirements may vary. The operator is given the option to include an extra cut or not as necessary.

The use of the block delete facility relies on a clear and concise relay of instructions between the part programmer and the machine operator. The machine operator must be left in no doubt as to what is required.

PROGRAM STOPS

Apart from the program stop that is automatically effected when the end of a program is reached, and which arrests all slide and spindle motions, there are two other situations where a halt in proceedings may need to be included in the part program.

The first of these is the point at which the operator is required to carry out some specific task directly associated with the machining program, such as resetting the work or replacing a tool. With word address programming this is normally achieved by programming M00. When such a stop is effected, it is essential that the operator knows exactly what has to be done before the program is reactivated. Some controls will allow for a man readable message to be read from the program and placed on the screen.

The second type of program stop is used when a halt in activity is not quite so critical, and the operator decides whether a stop is actually made. This type of stop is referred to as "optional" and will only take place if the operator has activated the optional stop button on the control console. The programmer may include an optional stop in the program whenever he or she considers it may be of value to the machine operator, such as when a dimensional check or an inspection of the tooling condition is appropriate. But quite often it is the operator who will edit into the program, via the control console, stops that will permit the solving of particular problems that have presented themselves during the machining process. The optional stop in a word address program is normally effected via a programmed M01.

In addition to the stops included in the part program, the operator has, of course, recourse to an emergency stop should the need arise.

CALCULATIONS

It could be argued that, in a well-organized CNC machining environment, the people responsible for the production of the detail drawings of the components to be machined should appreciate the needs of the part programmer and ensure that the drawings are dimensioned accordingly. For example, it is of considerable help when positional slide movements are to be programmed in absolute mode if all dimensions on the drawing are given in relation to a suitable datum. This is especially valuable when the programming technique involves conversational or manual data input, since the last thing a programmer wants to do is to interrupt his thought process in order to calculate unspecified dimensions.

However, whatever the ideal situation may be, it is almost certain that eventually the part programmer will be confronted with a detail drawing that does not cater to his or her requirements. He or she will then find it necessary to make calculations and add dimensions, and perhaps in some cases to completely redimension the drawing.

(The reader should differentiate between poor industrial practice and situations with which he or she may be deliberately confronted in a learning situation, where the objective will be to provide an understanding of the problems likely to be encountered in practice.)

Mention has already been made of the need for dimensions to be given in relation to a set datum when absolute programming is to be used. The opposite situation could also arise, whereby the dimensions are stated in relation to a datum but the programmer needs to program incremental slide moves. In this case the stated dimensions will need to be subdivided. The programmer should exercise caution in this particular situation, and ensure that such an approach is acceptable from a design point of view; minor errors on each of a series of incremental moves could, due to inaccuracies in calculations or round off, accumulate into a larger error that would be unacceptable.

There are other situations that are more complex than simply converting absolute dimensions to incremental and vice versa. Two of these in particular are the need to determine:

- (a) profile intersection points;
- (b) the location of arc centers.

However complex the profile or shape of a machined surface may appear to be, it can be broken down and defined geometrically as a number of intersecting straight lines or arcs or a combination of the two. To program appropriate machine slide movements, the programmer is required to determine this geometry, and translate production of the profile into a series of linear or circular movements.

Thus, to finish machine the profile shown in Figure 8.54 the following movements will be necessary:

- Move 1 Linear
- Move 2 Circular, clockwise
- Move 3 Linear
- Move 4 Circular, counterclockwise
- Move 5 Linear

If word address programming is being used, these moves can be described using the appropriate G code: G01, G02, G01, G03, and G01. It may be helpful to mark the drawing accordingly, as in the illustration.

The reader will already appreciate that when programming positional moves, whether they are linear or circular, the target position has to be numerically defined. In this particular example, because the component is dimensioned cor-

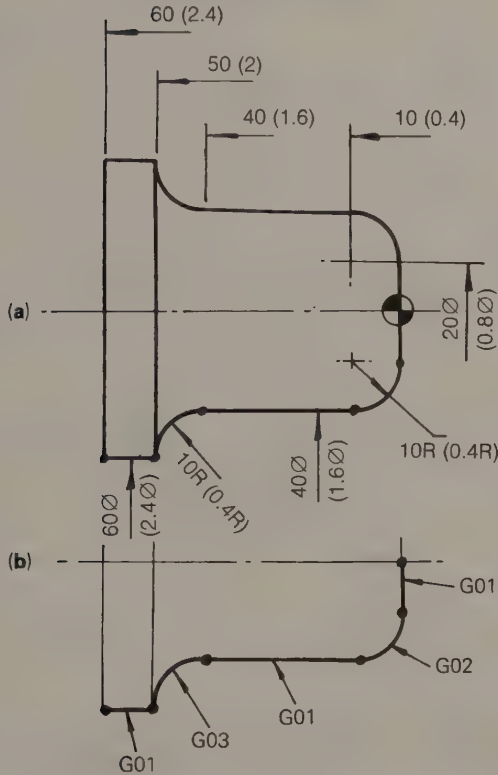


Figure 8.54 Geometric elements of a profile: (a) component detail; (b) profile definition. (Inch units are given in parentheses.)

rectly and the arcs are conveniently 90° , the target position, that is, the intersection points of the geometrical elements of the profile, are readily discerned. No further calculations are necessary.

Consider now the component shown in Figure 8.55(a). Although it is a relatively simple profile, from a programming point of view the drawing is not as helpful as it might be. The target position of the arc, that is, the point at which the circular move ends and the linear move commences, is not defined. Calculations are required as follows in order to determine the target position in the X and Z axes. (Only the metric unit calculation is given.)

In Figure 8.56:

$$\varnothing X_1 = 2(CB + 10) \text{ and } Z_1 = CD$$

In triangle ABC:

$$\angle ABC = 15^\circ \text{ and } AB = 12.5$$

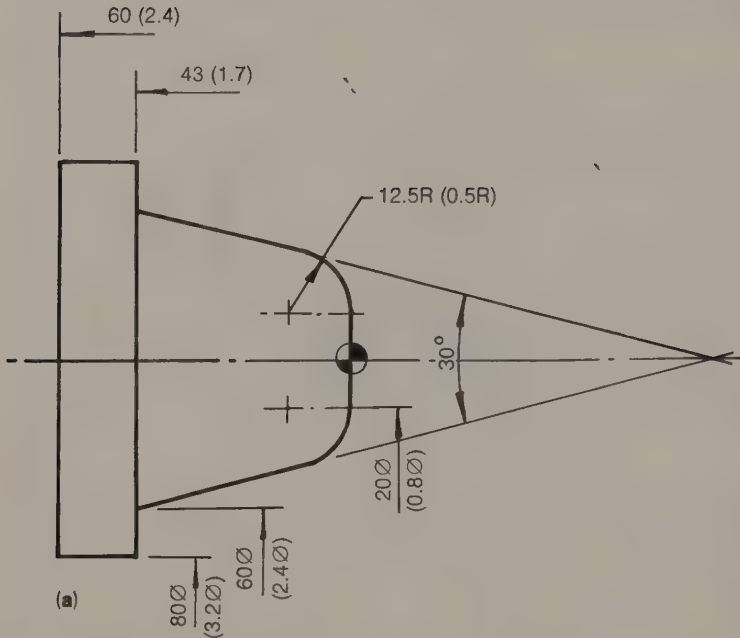


Figure 8.55 Component detail. (Inch units are given in parentheses.)

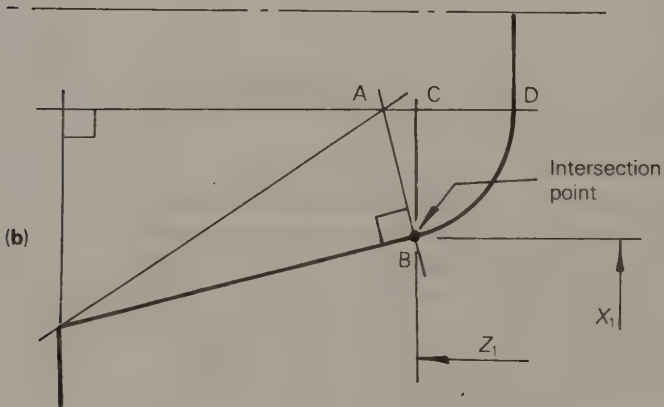


Figure 8.56 Profile intersection calculation.

1. To calculate CB:

$$\cos \angle ABC = \frac{CB}{AB}$$

$$\begin{aligned} CB &= \cos \angle ABC \times AB \\ &= \cos 15 \times 12.5 \\ &= 0.966 \times 12.5 \\ &= 12.074 \end{aligned}$$

$$\varnothing X_1 = 2(12.074 + 10) = 44.148$$

2. To calculate AC:

$$\sin \angle ABC = \frac{AC}{AB}$$

$$\begin{aligned} AC &= \sin \angle ABC \times AB \\ &= \sin 15 \times 12.5 \\ &= 0.259 \times 12.5 \\ &= 3.235 \end{aligned}$$

$$\begin{aligned} \text{Since } Z_1 &= CD, \text{ then } Z_1 = AD - AC \\ &= 12.5 - 3.235 \\ &= 9.265 \end{aligned}$$

Thus, the target position, with the X value being programmed as a diameter, is $X44.148 \ Z-9.265$. These values would need to be included in the part program.

This particular calculation is fairly typical of the situations the part programmer has to deal with. A similar situation presents itself in the profile shown in Figure 8.57(a) where one radius blends with another. The problem is: where does one radius end and the second one start? Calculations are necessary to determine the location of point P in the X and Y axes as indicated in Figure 8.57(b). The reader may like to consider the solution to the problem. (Answers: 47.81 mm and 71.25 mm, respectively.)

This type of profile also presents the second type of calculation referred to earlier, namely, determining the location of arc centers.

From the previous information the reader will recall that when circular arcs are programmed using word address programming one of three techniques may be involved. All require the target positions to be identified, but the radius definition varies. The first involves defining the arc center in relation to the program datum, and the second requires the arc center to be defined in relation to the arc starting position.

Using the first method, the center of the 30 mm radius arc is easily determined from the dimensions already on the drawing. But the location of the center of the 50 mm radius is not so straightforward and a calculation is required.

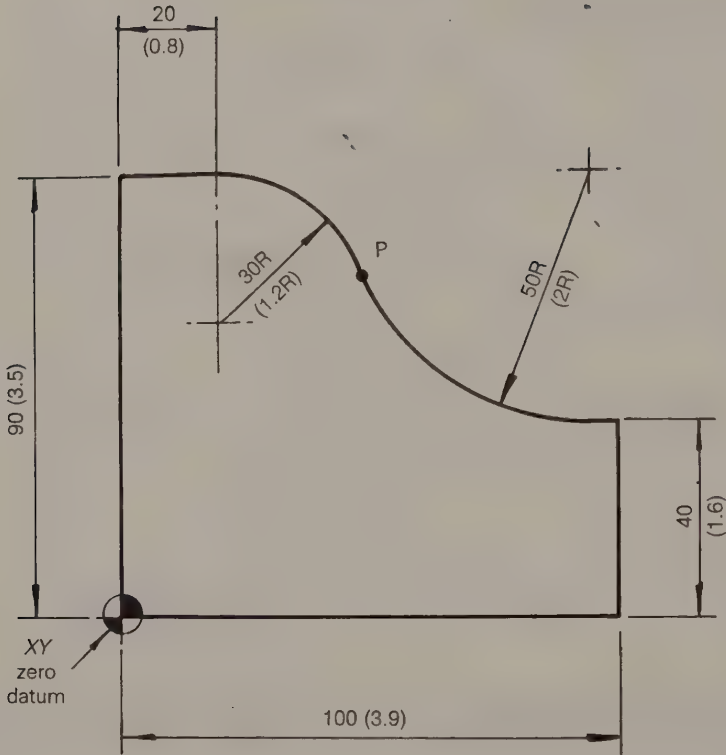


Figure 8.57 (a) Profile detail. (Inch units are given in parentheses.)

Using the second method of circular interpolation the definition of the arc centers in relation to the start points again presents a problem as far as the 50 mm radius is concerned, and a calculation will be necessary before the program can be written.

Exercises involving the calculation of profile intersection points and arc centers are included in this chapter's section titled "Part Programming Calculations."

TOOL PATHS

A prime objective of the part programmer should be to ensure that a component is machined in the shortest possible time. Earlier in the text reference was made to the way a well-planned sequence of machining operations can contribute to this objective. But often within each individual machining sequence there is room for further efficiency, which results in time saving.

Consider the drilling of the series of five holes in the component shown in Figure 8.58. Two sequences in which the holes might be drilled are indicated. Which sequence would be the quicker?

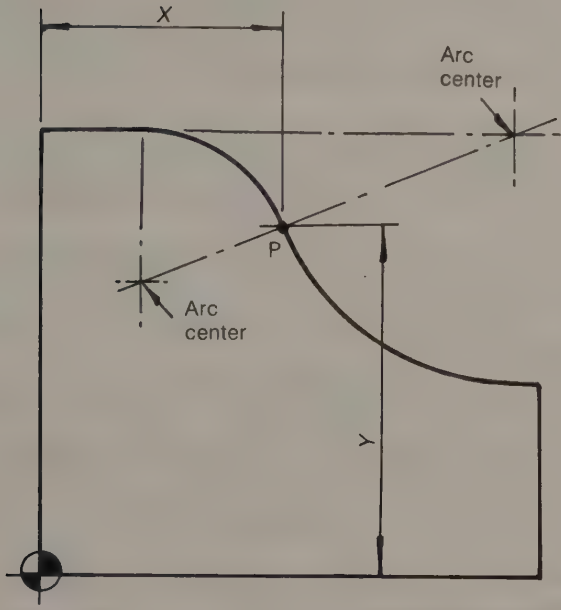


Figure 8.57 (b) Required profile intersection dimensions.

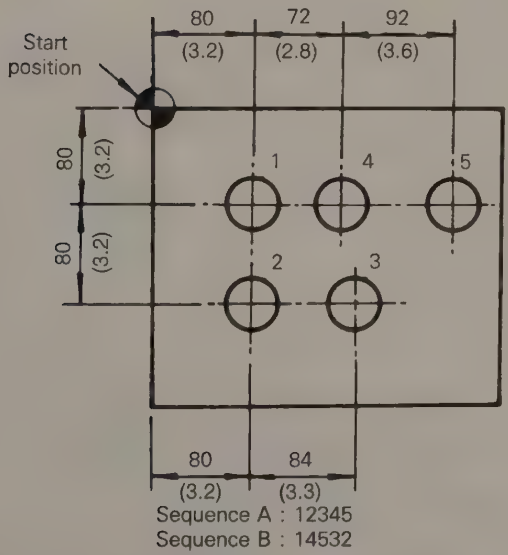


Figure 8.58 Alternative drilling sequences. (Inch units are given in parentheses.)

The actual operation of drilling the holes, that is, movement in the Z axis, would be identical in both cases. Therefore, any saving that can be achieved must be by reducing the total length of the positioning moves, and therefore the time taken.

Providing the detail drawing is reasonably accurately drawn a simple rule measurement check may suffice to determine the shorter route. Applying this technique to this particular example will reveal that the second sequence is quicker than the first.

The need to give careful consideration to tool paths is also important during stock removal operations. This is particularly so when there are no stock removal canned cycles available within the control system, or if they cannot be utilized in a particular situation.

Consider the removal of stock, or area clearance as it is also known, in order to machine the step shown in Figure 8.59.

If a pocket milling cycle is available on the control system of the machine, this could be used, the missing sides of the "pocket" being indicated by the dotted line. Use of the cycle would ensure that efficient tool paths are employed.

If such a cycle is not available, then the matter becomes a little more complex. The process of producing the step will involve programming a series of linear moves, with careful attention being given to providing an appropriate cutter overlap to ensure a clean face. The lengths of relative cutter travel will also have to ensure a uniform amount of metal is left for a finish pass along

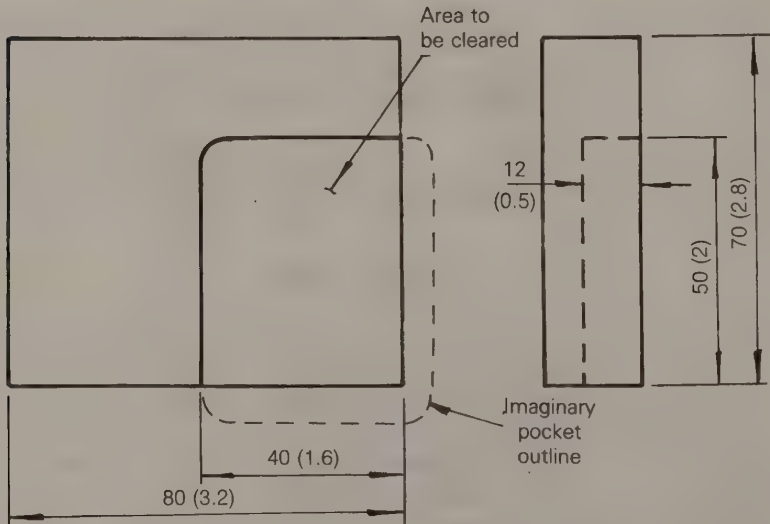


Figure 8.59 "Pocket" detail. (Inch units are given in parentheses.)

the profile. The programmer should also ensure that the cutter paths used are the shortest and therefore the quickest.

Similar problems often present themselves during turning operations. Figure 8.60 shows a typical example.

There is no short cut when solving this type of problem. After a little experience of dealing with situations of this nature, the trainee programmer soon comes to appreciate the value of canned cycles, which reduce the amount of machining dealt with in this manner to a minimum.

If the part programmer is confronted with machining situations such as these, he or she will have to resort to drawing the profile, preferably to an enlarged scale, and then imposing appropriate tool paths on the drawing. In the case of milling examples it may be necessary to draw circles indicating the cutter diameter. The milled step referred to above, when dealt with in this way, is shown in Figure 8.61. Having decided on the most suitable tool paths (which may take a number of attempts), the slide movements may be dimensionally determined by carefully scaling the drawing or through mathematical calculation.

An alternative approach is to reproduce the profile on graph paper, as shown in Figure 8.62, in which case the graduated lines on the graph paper can be used to determine the dimensional value of the necessary moves.

Exercises involving the determination of cutter paths are included in subsequent examples.

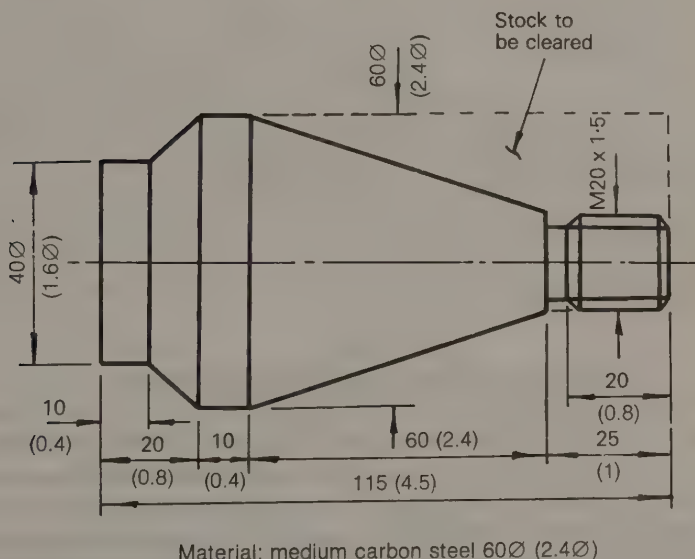


Figure 8.60 Component requiring excess stock removal. (Inch units are given in parentheses.)

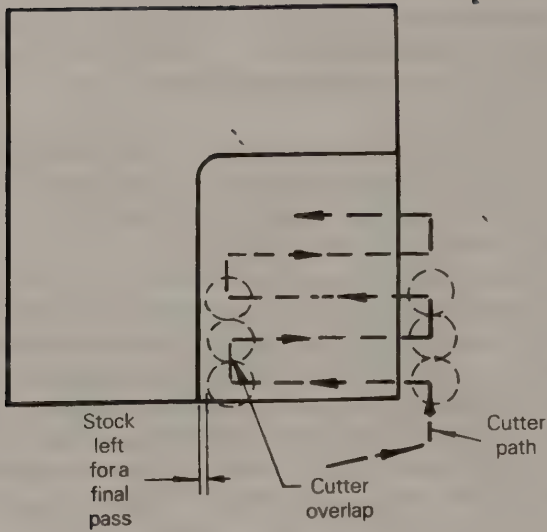


Figure 8.61 *Determination of cutter path to mill a step.*

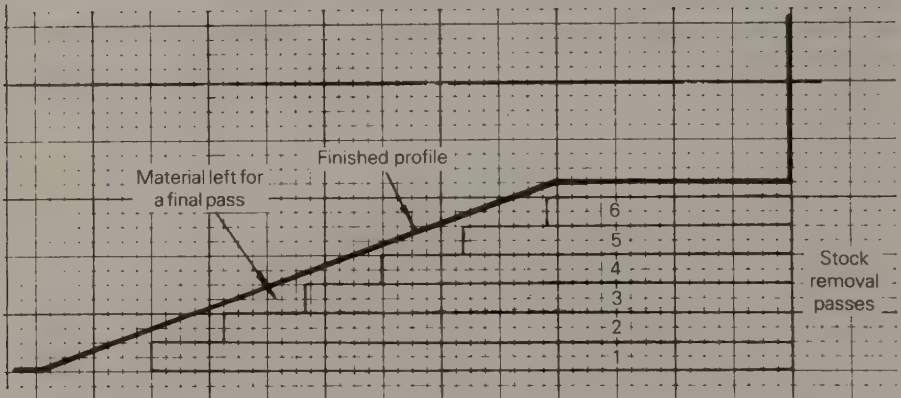


Figure 8.62 *Determination of tool paths when turning.*

PROGRAM LISTING AND PROVING

Before starting to create a part program listing, all the various facets of competent part programming discussed so far should have received due attention.

The sequence of operations, together with the tooling and work-holding techniques to be employed, should be documented. Appropriate speeds and feed rates should have been determined and all the necessary calculations affecting slide movements must be complete.

Having reached this stage the way is clear to list the program, and this requires the programmer to be conversant with the machine programming language.

To become fully proficient with a particular programming system takes time and practice. As with most things it is a case of starting with relatively simple tasks and gradually progressing to more complex examples. If you are a student, or perhaps undergoing training in an industrial establishment, it is almost certain that your course work will be structured in this way.

Competent part programming demands a logical approach and a high degree of concentration and care when actually listing the program. Mistakes are easily made and can have disastrous results, although fortunately most mistakes can be discovered and rectified before machining takes place.

Programs may be listed on appropriate forms, or on plain paper or they can be entered into a computer and listed on the display screen. Programs initially handwritten can, of course, also be entered into a computer and visually displayed.

The use of a computer for program listing is often coupled with the facility to prove the program using animated computer graphics. This involves, in effect, "machining" the component on the screen.

The effectiveness of proving programs in this way will depend on the sophistication of the software available. The simplest software will usually highlight major errors such as movement occurring in the wrong direction or a lathe tool crashing into a chuck, while the more complex will also indicate errors relating to speeds and feeds and even the absence of a coolant supply.

Ultimately, the part program will be entered into the machine control unit, but this may also involve computer graphics. Figure 8.63 shows a controller that includes a built-in visual display. As the program is entered the CRT screen will display the geometric profile of the part and the programmed cutter paths and thus confirm, or otherwise, the validity of the data input. The illustration relates to a program written for the part detailed in Figure 8.64. An enlargement of the CRT display is shown in Figure 8.65 where the component profile can be more readily defined.

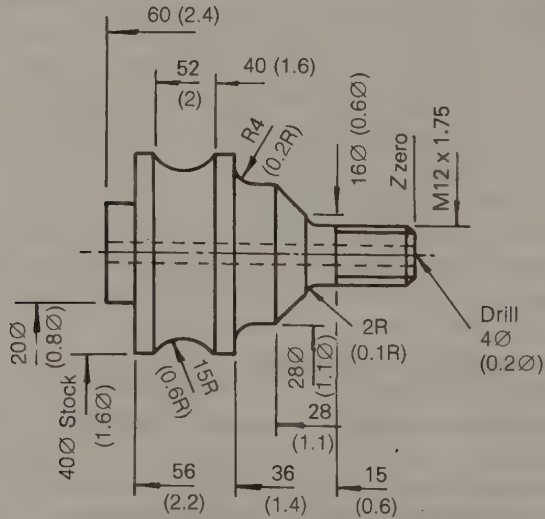
A large number of machines currently in use do not have the benefit of built-in computer graphics, and if off-line computer graphics proving facilities are not available, then the proving of the part program must take the form of a test run or a dry run or both.

The test run is basically a check that the data input is valid, that is, that the machine is capable of responding to the data entries included in the program. Data errors are usually indicated by a displayed message. No slide movement takes place during the test run.

The dry run procedure also excludes metal cutting, but, with this checking procedure, slide movements occur at a rapid rate of traverse. This test ensures that the intended machine movements are occurring, and that they will result in the machined features required.



Figure 8.63 A machine controller with built-in CRT to facilitate program proving.



Material: aluminum alloy 40Ø (1.6Ø)

Figure 8.64 Component detail. (Inch units are given in parentheses.)

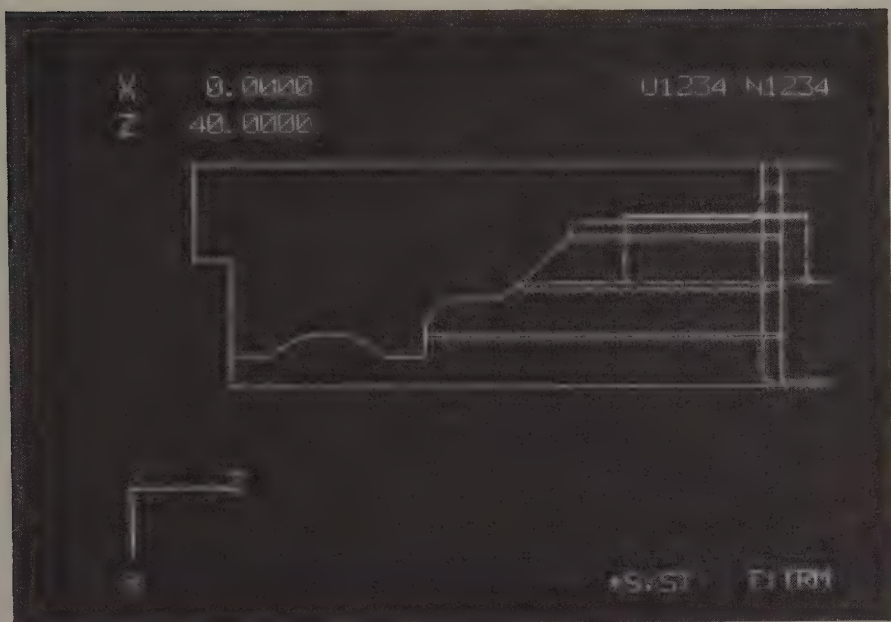


Figure 8.65 An enlargement of the CRT display shown in Figure 8.63.

The most common test available, even on the simplest of machines, is to run the program through one block at a time with reduced programmed feed rates, but without the job material being in position so that no machining actually takes place—block by block prove out. This prove out method is usually followed by a block by block run with part material and reduced feed rates before an automatic run is attempted.

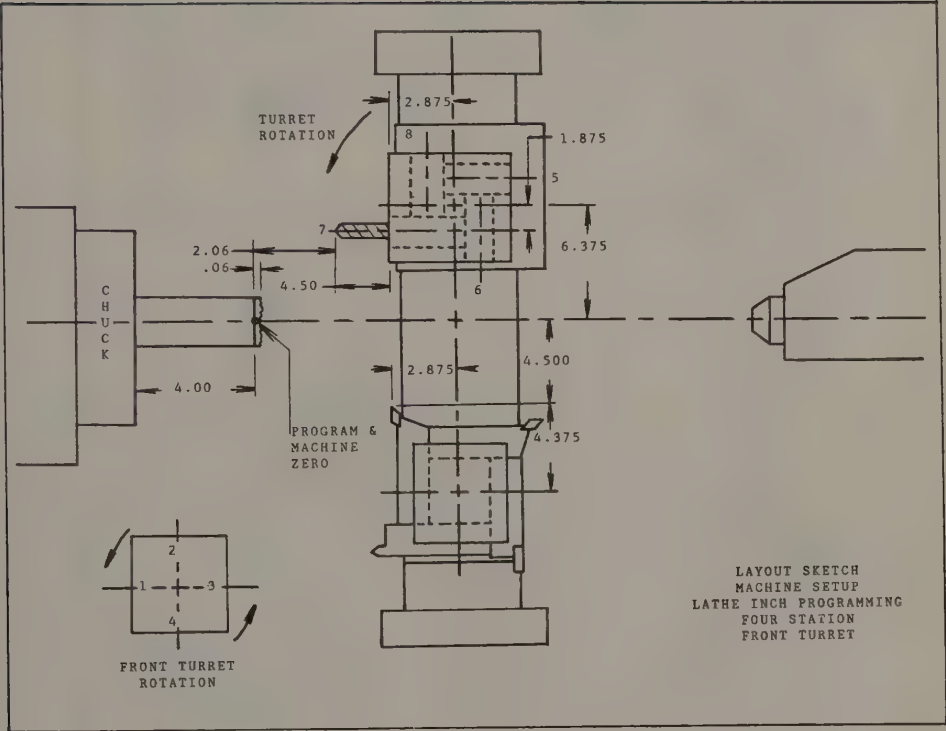
Tests of this nature carried out on the machine may or may not be the responsibility of the programmer, although he or she will soon be involved if any errors are indicated.

Finally, there is the need, particularly in industrial situations, to record the program for future use. In its simplest form, storage can be a handwritten version of the proved program. Alternatively, it may be in the form of a perforated tape or be recorded on a magnetic tape or disk.

Whatever the storage medium, it must be remembered that people looking to reuse the program at a later date will also need information relating to tooling and workholding. This information is as critical as the part program and must also be carefully filed for future reference.

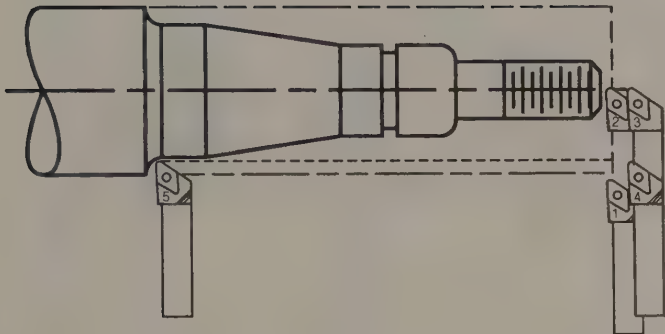
MANUAL PART PROGRAMMING EXAMPLES

The programming examples that follow were prepared to show the calculation and writing of generic manual part programs. Examples follow for both a lathe



TOOL SHEET

PART NO. NC-2118702		MATERIAL ALUM.		OPERATION 20		PROGRAMMED BY T. CRANDELL	
PART NAME PROJECT 1		B/P CHANGE DATE 9-21-87		DATE 5-5-89		SHEET 1 OF 1	
TOOL	TOOL	TOOL LENGTH X		TOOL LENGTH Z		SPEED	
SEQ. NUMBER	DESCRIPTION	PROG.	ACT.	#	PROG.	R P M	FEED
T01	55 DEG. DIAMOND TURNING TOOL (RGH) .031 TOOL NOSE RADIUS Valenite Holder MDJNR-13-4 Insert DNMP432E VN5	4.375		2.875		960 to 2000	.010 IPR
T02	55 DEG. DIAMOND TURNING TOOL (FIN) .031 TOOL NOSE RADIUS Valenite Holder MDJNR-13-4 Insert DNMP432E VN5	4.375		2.875		2000	.005 IPR
T03	.125 GROOVE TOOL Valenite Holder SD-IMR-16-3 Insert TMC32NG VN5	4.375		2.875		1000	.005 IPR
T04	60 DEG. THREAD TOOL Valenite Holder SD-IMR-16-3 Insert TMC32NV VC7	4.375		2.875		2000	.077 IPR
T07	.250 DIA. HSS DRILL	1.875		4.500		2000	.006 IPR



COORDINATE CALCULATIONS
LATHE INCH PROGRAMMING

ROUGH FACE & FIRST ROUGH TURN

POINT

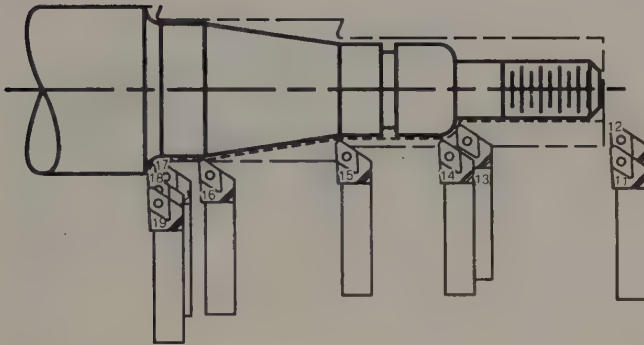
CODE

X COORDINATE

Z COORDINATE

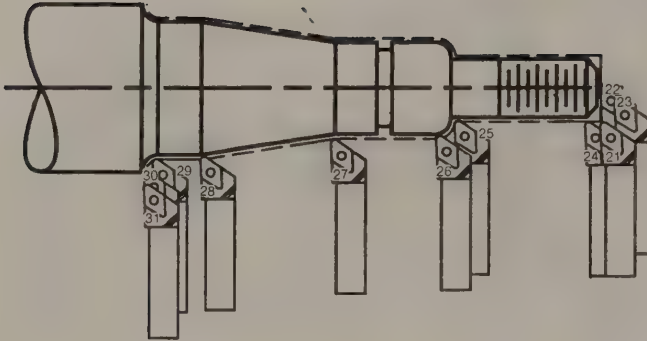
CALCULATION

P1	X .725	Z .030	X .625 Stock Radius + .100 Clearance Z .030 Finish Stock
P2	X-.050	Z .030	X-.030 Tool Radius - .020 Cut Past Center Z .030 Finish Stock
P3	X-.050	Z .050	X-.030 - .020 Z .030 Finish Stock + .020 Clearance
P4	X. 525	Z .050	X .500 Part Radius + .025 Stock
P5	X .525	Z-3.406	$Z-3.375 - \sqrt{.069^2 - .069^2} - .031$ Formula #1 Appendix E



SEMI FINISH TURN

POINT CODE	X COORDINATE	Z COORDINATE	CALCULATION
P11	X .450	Z .050	Z .030 Finish Stock + .020 Clearance
P12	X .265	Z .050	X .250 Part Radius + .015 Finish Stock
P13	X .265	Z-1.1164	Z-1.125 - .125 + .1646 - .031 Formula #3 Appendix E
P14	X .390 I.046	Z-1.281 K.1646	X .375 Part Radius + .015 Finish Stock Z-1.250 Radius Center - .031 Tool Radius
P15	X .390	Z-2.0283	Z-2.0 - [.031 - (Tan. $\frac{6.7129^\circ}{2}$ X (.031 + .015))])] Formula #2 Appendix E
P16	X .515	Z-3.0903	X .500 Part Radius + .015 Finish Stock Z-2.0283 Pos. 15 - 1.062 Taper Length
P17	X .515	Z-3.406	Z-3.500 Part Dim. + .125 Radius - .031 Tool Radius
P18	X .594 I.079	Z-3.485 KO	X .625 Part Radius - .031 Tool Radius Z-3.500 Part Dim. + .015 Finish Stock
P19	X .750	Z-3.485	X .625 Part Radius + .125 Clearance



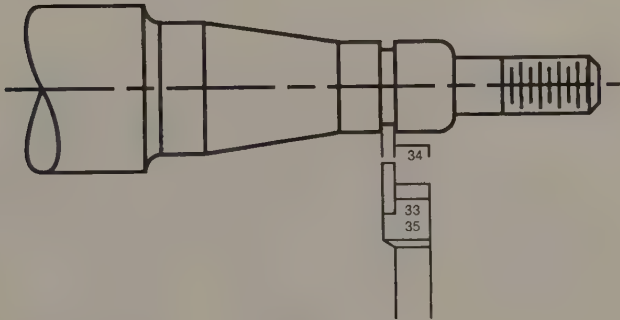
FINISH FACE AND FINISH TURN

POINT

CODE	X COORDINATE	Z COORDINATE	CALCULATION
P20	TOOL	CHANGE	POSITION
P21	X .365	Z 0	X .250 Part Radius + .015 Stock + .100 Clearance Z 0 Finish Datum
P22	X-.050	Z 0	X 0 Datum - .031 Tool Radius - .019 Past Center
P23	X .0718	Z .100	X .250 Part Radius - .06 Cham. - .10 Clearance +.0128 Formula #4 Appendix E -.031 Tool Radius Z 0 Datum + .100 Clearance
P24	X .250	Z .0782	X .250 Part Radius Z-.060 Chamfer + .0128(formula 4) -.031 Tool Radius
P25	X .250	Z-1.1281	Z-1.125 - .125 + .1529 - .031 Formula #3 No Stock Appendix E
P26	X .375 I.031	Z-1.281 K.1529	X .375 Part Radius Z-1.125 - .125 radius - .031 Tool Radius
P27	X .375	Z-2.0292	Z-2.0 -[.031-(Tan. $\frac{6.7129^\circ}{2}$ X.031)] Formula #2 No Stock Appendix E
P28	X .500	Z-3.0912	X .500 Part Radius Z-2.0292 Pos. 27 - 1.062 Taper Length

POINT
CODE X COORDINATE Z COORDINATE CALCULATION

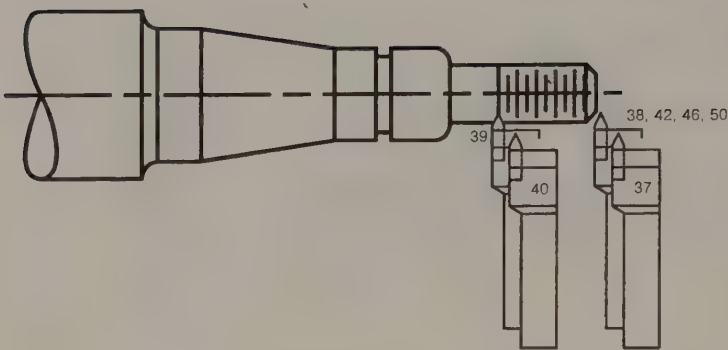
P29	X.500	Z-3.406	Z-3.500 Part Dim. + .125 Radius - .031 Tool Radius
P30	X .594 I.094	Z-3.500 K0	X .625 Part Radius - .031 Tool Radius Z-3.500 Part Dim.
P31	X .750	Z-3.500	X .625 Part Radius + .125 Clearance
P32	TOOL	CHANGE	POSITION



CUT GROOVE

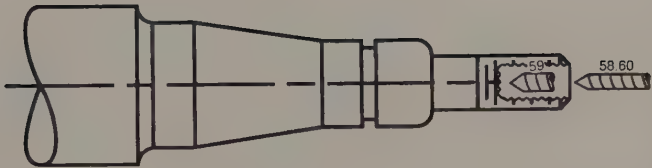
POINT
CODE X COORDINATE Z COORDINATE CALCULATION

P33	X .475	Z-1.687	X .375 Part Radius + .100 Clearance Z-1.561 Part Dim. - .125 Groove Width
P34	X .3125	Z-1.687	X .3125 Groove Radius
P35	X .475	Z-1.687	X .375 Part Radius + .100 Clearance
P36	TOOL	CHANGE	POSITION



THREAD CUTTING

POINT CODE	X COORDINATE	Z COORDINATE	CALCULATION
P37	X .350	Z .100	X .250 Part Radius + .100 Clearance Z 0 Finish Datum + .100 clearance
P38	X .2284	Z .0876 K.0769	X .250 Part Radius - .0216 Thread In Feed Z .100 Clearance - .0124 Thread In Feed K 1 ÷ 13 (Thread Pitch)
P39	X .2284	Z-.750	Z-.750 Thread Length Dim.
P40	X .350	Z-.750	X .250 Part Radius + .100 Clearance
P42	X .2166	Z .0808	X .250 Part Radius - .0334 Thread In Feed Z .100 Clearance - .0192 Thread In Feed
P46	X .2076	Z .0756	X .250 Part Radius - .0424 Thread In Feed Z .100 Clearance - .0244 Thread In Feed
P50	X .200	Z .0714	X .250 Part Radius - .050 Thread In Feed Z .100 Clearance - .0286 Thread In Feed
	TOOL	CHANGE	POSITION



DRILL .250 DIAMETER HOLE

POINT CODE	X COORDINATE	Z COORDINATE	CALCULATION
P58	X 0	Z .100	X 0 Datum Z 0 Datum + .100 Clearance
P59	X 0	Z -.575	Z -.500 Hole Depth - .075 Drill Point .3 X Dia. Drill Point Allowance
P60	X 0	Z .100	X 0 Datum Z 0 Datum + .100 Clearance
P61	TOOL	CHANGE	POSITION

Lathe—Inch Program (Example 1)

```

%           Rewind stop code
N0010 G70   Inch programming
N0020 G97   rpm spindle speed programming
N0030 G95   ipr feedrate programming
N0040 G90   Absolute coordinate programming
N0050 G00 T00   Clear tool offsets
N0060 T0600   Index rear turret to empty station
N0070 T0101 S960 M03   Index tool 1 to cutting position (rough face & turn),
                        start spindle
N0080 G92 X4.5 Z6.56   Preset X & Z axes
N0090 G00 X.725 Z.03 M08   Position tool to rough face, turn coolant on (P1)
N0100 G01 X-.05 F.01   Rough face part leaving 0.030 (P2)
N0110 G00 Z.050   Retract from face (P3)
N0120 X.525   Position X for first rough turn (P4)
N0130 G01 Z-3.406 F.01   Feed for first rough turn (P5)
N0140 G00 X.575 Z-3.356   Retract X & Z axes (P6)
N0150 Z.05 S1132   Rapid to face of part (P7)
N0160 X.4   Position X for second rough turn (P8)
N0170 G01 Z-2.0277 F.01   Feed for second rough turn (P9)
N0180 G00 X.450 Z-1.9777   Retract X & Z axes (P10)
N0190 Z.05 S2000   Rapid to face of part (P11)
N0200 X.265   Position X for semifinish pass (P12)
N0210 G01 Z-1.1164   Feed tangent to convex radius (P13)
N0220 S1538   Change rpm
N0230 G03 X.39 Z-1.281 I.046 K.1647   Contour convex radius (P14)
N0240 G01 Z-2.0283   Feed tangent to taper (P15)
N0250 S1200   Change rpm
N0260 X.515 Z-3.0903   Feed up taper (P16)
N0270 Z-3.406   Feed tangent to concave radius (P17)
N0280 S1010   Change rpm
N0290 G02 X.594 Z-3.485 I.079 K0   Contour concave radius (P18)
N0300 G01 X.750   Feed off part (P19)
N0310 G00 X4.5 Z6.56 T0000   Rapid to tool change position—cancelling tool
                        offset (P20)
N0320 G92 X0 Z0   Cancel axis presets
N0330 G00 T0202   Index finish facing and turning tool into cutting position
N0340 G90 S2000 M03   Set finishing rpm
N0350 G92 X4.5 Z6.56   Axis presets
N0360 G00 X.365 Z0   Rapid position to start finish face (P21)
N0370 G01 X-.05 F.005   Finish face part (P22)
N0380 G00 X.0718 Z.100   Retract in Z and position for chamfer (P23)
N0390 G01 X.25 Z.0782   Turn chamfer (P24)
N0400 Z-1.1281   Finish turn to convex radius (P25)
N0410 S1600   Change rpm
N0420 G03 X.375 Z-1.281 I.031 K.1529   Contour convex radius (P26)
N0430 G01 Z-2.0292   Finish turn tangent to taper (P27)
N0440 S1200   Change rpm
N0450 X.5 Z-3.0912   Finish turn taper (P28)
N0460 Z-3.406   Finish turn tangent to concave radius (P29)
N0470 S1010   Change rpm

```

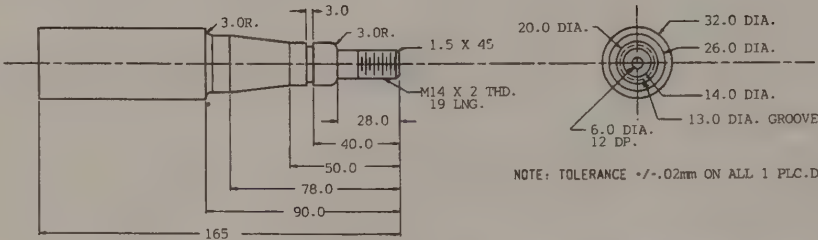
```

N0480 G02 X.594 Z-3.5 I.094 K0      Contour concave radius (P30)
N0490 G01 X.750      Feed away from part (P31)
N0500 G00 X4.5 Z6.56 T0000 M09      Rapid to tool change position cancelling tool
                                     offsets and turning off coolant (P32)
N0510 G92 X0 Z0      Cancel axis presets
N0520 G00 T0303      Index grooving tool into cutting position
N0530 G90 S1000 M03      Start spindle
N0540 G92 X4.5 Z6.56      Preset axes
N0550 G00 X.475 Z-1.687 M08      Position in X & Z to cut groove (P33)
N0560 G01 X.3125 F.005      Cut groove to depth (P34)
N0570 G04 F05      Dwell for 5 seconds
N0580 G00 X.475 M09      Retract from groove, turning coolant off (P35)
N0590 X4.5 Z6.56 T0000      Rapid to tool change position cancelling tool offset
                                     (P36)
N0600 G92 X0 Z0      Cancel axis presets
N0610 G00 T0404      Index threading tool into cutting position
N0620 G90 S2000 M03      Start spindle
N0630 G92 X4.5 Z6.56      Preset axes
N0640 G00 X.35 Z.1 M08      Position in X & Z to cut thread (P37)
N0650 X.2284 Z.0876      First thread in feed (P38)
N0660 G33 Z-.75 K.0769      Cut first thread pass (P39)
N0670 G00 X.35      Retract X (P40)
N0680 Z.1      Rapid to end of part (P41)
N0690 X.2166 Z.0808      Thread in feed for second pass (P42)
N0700 G33 Z-.75 K.0769      Cut second pass (P43)
N0710 G00 X.35      Retract X (P44)
N0720 Z.1      Rapid to end of part (P45)
N0730 X.2076 Z.0756      Thread in feed for third pass (P46)
N0740 G33 Z-.75 K.0769      Cut third pass (P47)
N0750 G00 X.35      Retract X (P48)
N0760 Z.1      Rapid to end of part (P49)
N0770 X.2 Z.0714      Thread in feed for finish pass (P50)
N0780 G33 Z-.75 K.0769      Cut finish pass (P51)
N0790 G00 X.35      Retract X (P52)
N0800 Z.1      Rapid to end of part (P53)
N0810 X.2 Z.0714      Thread in feed for tool pressure pass (P54)
N0820 G33 Z-.75 K.0769      Cut tool pressure pass (P55)
N0830 G00 X.35 M09      Retract X (P56)
N0840 X4.5 Z6.56 T0000      Rapid to tool change cancelling tool offsets (P57)
N0850 G92 X0 Z0      Cancel axis preset
N0860 G00 T0707      Index 0.250 diameter drill into cutting position
N0870 G90 S2000 M03      Start spindle
N0880 G92 X-4.5 Z2.06      Preset axes
N0890 G00 X0 Z.1 M08      Position to drill (P58)
N0900 G01 Z-.575 F.006      Drill to depth (P59)
N0910 G00 Z.1 M09      Retract drill (P60)
N0920 X-4.5 Z2.06 T0000      Rapid to tool change cancelling tool offsets (P61)
N0930 G92 X0 Z0      Cancel axis presets
N0940 M02      End of program

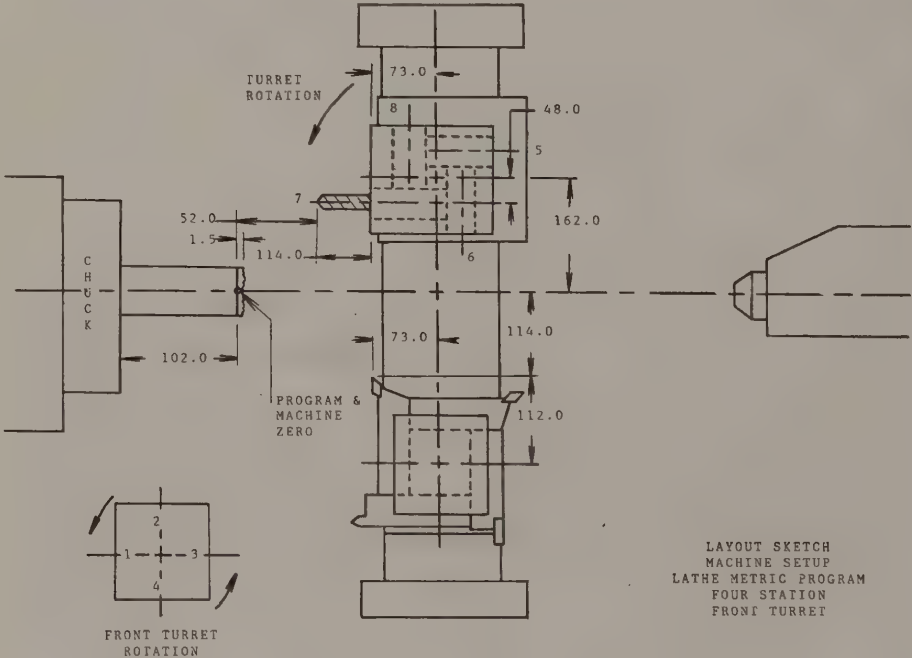
```

Example 2: Lathe Metric Programming

LATHE - METRIC PROGRAMMING



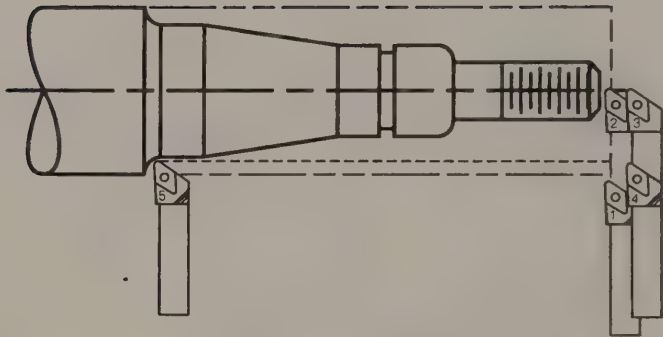
ALUM.		F. CRANDELL	SHELDON M/C LATHE PROJECT # 1		
UNLESS OTHERWISE SPECIFIED TOLERANCES IN DIMENSIONS 2 PLACES - .01, 3 PLACES - .005 ANGLES - 1 DEG. UNLESS ALL DIMENSIONS ARE METRIC NOTE: ALL DIMENSIONS ARE METRIC		DATE	REVISED BY	REV.	
		NONE	B	NC-2118702	A



TOOL SHEET

LATHE METRIC PROGRAMMING

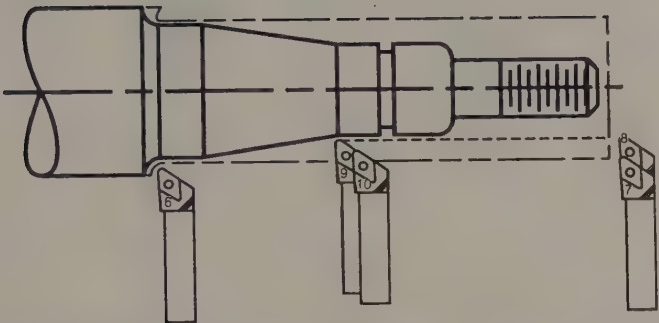
PART NO. NC-2118702		MATERIAL ALUM.		OPERATION 20		PROGRAMMED BY T. CRANDELL	
PART NAME PROJECT 1		B/P CHANGE DATE		DATE		SHEET 1 OF 1	
TOOL SEQ. NUMBER	TOOL DESCRIPTION	TOOL LENGTH X		TOOL LENGTH Z		OPERATION DESCRIPTION	FEED IN/MM CODE
		PROG.	ACT.	#	PROG.	ACT.	
T01	55 DEG. DIAMOND TURNING TOOL (RGH) 0.8 mm TOOL NOSE RADIUS Valenite Holder MDJNR-13-4 Insert DNMF432E VN5	112.0			73.0	RGH. FACE & RGH. TURN	.25 MMPR
T02	55 DEG. DIAMOND TURNING TOOL (FIN) 0.8 mm TOOL NOSE RADIUS Valenite Holder MDJNR-13-4 Insert DNMF432E VN5	112.0			73.0	FIN. FACE & FIN. TURN	.12 MMPR
T03	3.0 GROOVE TOOL Valenite Holder SD-TMR-16-3 Insert TMMC32NG VN5	112.0			73.0	3.0 GROOVE	.12 MMPR
T04	60 DEG. THREAD TOOL Valenite Holder SD-TMR-16-3 Insert TMMC32NV VC7	112.0			73.0	M14X2 THREAD	2.0 MMPR
T07	6.0mm DIA. HSS DRILL	48.0			114.0	DRILL 6.0mm DIA.	.15 MMPR



COORDINATE CALCULATIONS
LATHE METRIC PROGRAMMING

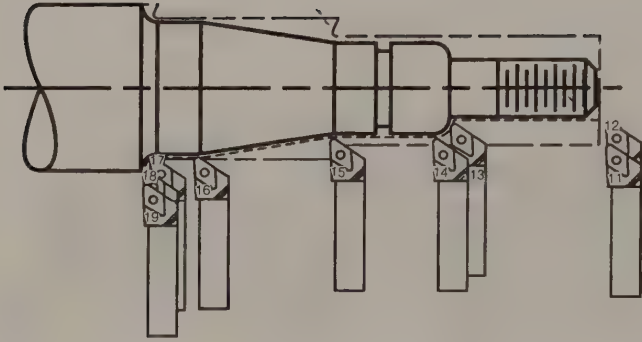
ROUGH FACE & FIRST ROUGH TURN

POINT CODE	X COORDINATE	Z COORDINATE	CALCULATION
P1	X 18.5	Z .75	X 16.0 Stock Radius + 2.5 Clearance Z .75 Finish Stock
P2	X-1.3	Z .75	X-.8 Tool Radius - .5 Cut Past Center Z .75 Finish Stock
P3	X-1.3	Z 3.25	X-.8 - .5 Z .75 Finish Stock + 2.5 Clearance
P4	X 13.6	Z 3.25	X 13.0 Part radius + .6 Stock
P5	X 13.6	Z-87.8	$Z-87.0 - \sqrt{1.6^2 - 1.6^2} - .8$ Formula #1 Appendix E



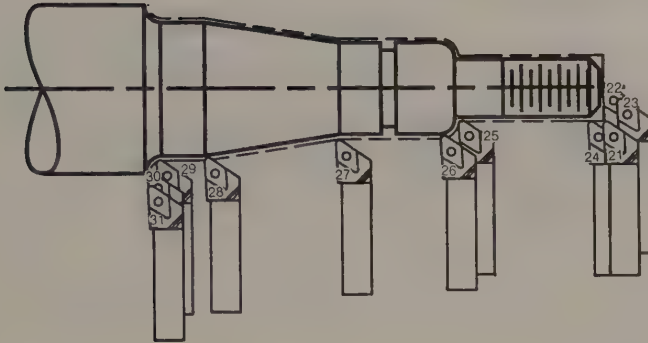
SECOND ROUGH TURN

POINT CODE	X COORDINATE	Z COORDINATE	CALCULATION
P6	X 14.8	Z-86.6	X 13.6 Radius + 1.2 Clearance Z-87.8 Pos."5" + 1.2 Clearance
P7	X 14.8	Z 1.25	Z .75 Finish Stock + .5 Clearance
P8	X 10.6	Z 1.25	X 10.0 Part Radius + .6 Stock
P9	X 10.6	Z-50.72	$Z-50.0 - [.8 - (\tan \frac{6.7129^\circ}{2} \times (.8 + .6))]$ Formula #2 Appendix E
P10	X 11.8	Z-49.52	X 10.6 Radius + 1.2 Clearance Z-50.72 Pos."9" + 1.2 Clearance



SEMI FINISH TURN

POINT CODE	X COORDINATE	Z COORDINATE	CALCULATION
P11	X 11.8	Z 1.25	Z .75 Finish Stock + .5 Clearance
P12	X 7.4	Z 1.25	X 7.0 Part Radius + .4 Finish Stock
P13	X 7.4	Z-27.78	Z-28.0 - 3.0 + 4.02 - .8 Formula #3 Appendix E
P14	X 10.4 11.2	Z-31.8 K4.02	X 10.0 Part Radius + .4 Finish Stock Z-31.0 Radius Center - .8 Tool Radius
P15	X 10.4	Z-50.73	Z-50 -[.8-(Tan. $\frac{6.7129^\circ}{2}$ X (.8+.4))] Formula #2 Appendix E
P16	X 13.4	Z-78.73	X 13.0 Part Radius + .4 Finish Stock Z-50.73 Pos. 15 - 28.0 Taper Length
P17	X 13.4	Z-87.8	Z-90 Part Dim. + 3.0 Radius - .8 Tool Radius
P18	X 15.2 11.8	Z-89.6 K0	X 16.0 Part Radius - .8 Tool Radius Z-90 Part Dim. + .4 Finish Stock
P19	X 18.5	Z-89.6	X 16.0 Part Radius + 2.5 Clearance

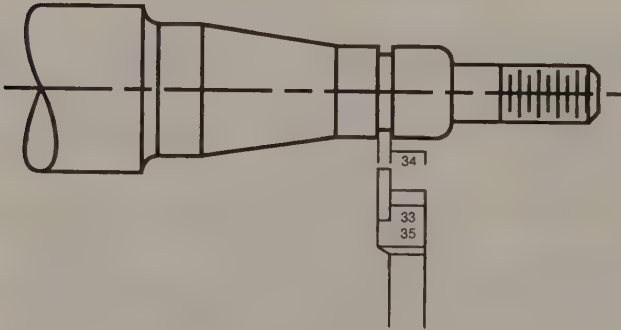


FINISH FACE AND FINISH TURN

POINT
CODE X COORDINATE Z COORDINATE CALCULATION

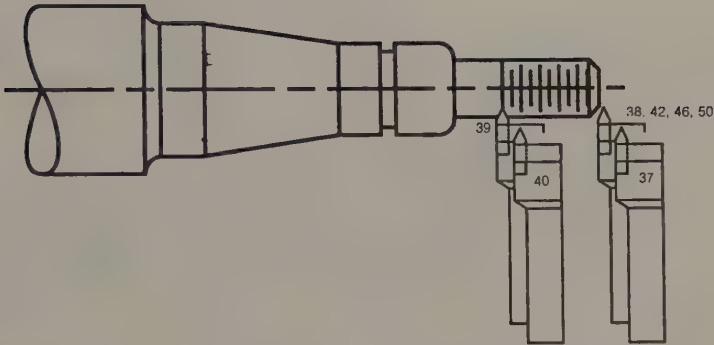
P20	TOOL	CHANGE	POSITION
P21	X 9.9	Z 0	X 7.0 Part Radius + .4 Stock + 2.5 Clearance Z 0 Finish Datum
P22	X-1.3	Z 0	X 0 Datum - .8 Tool Radius - .5 Past Center
P23	X 2.53	Z 2.5	X 7.0 Part Radius - 1.5 Cham. - 2.5 Clear + .33 Formula #4 Appendix E - .8 Tool Radius Z 0 Datum + 2.5 Clearance
P24	X 7.0	Z-1.97	X 7.0 Part Radius Z-1.5 Chamfer + .33 (formula 4) - .8 Tool Radius
P25	X 7.0	Z-25.39	Z-28.0 - .3 + 3.71 - .8 Formula #3 No Stock Appendix E
P26	X 10.0 I.8	Z-31.8 K3.71	X 10.0 Part Radius Z-28.0 - 3.0 Radius - .8 Tool Radius
P27	X 10.0	Z-50.75	Z-50.0 - [.8 - (Tan. $\frac{6.7129^\circ}{2}$ x .8)] Formula #2 No Stock Appendix E
P28	X 13.0	Z-78.75	X 13.0 Part Radius Z-50.75 Pos. 27 - 28.0 Taper Length

POINT CODE	X COORDINATE	Z COORDINATE	CALCULATION
P29	X 13.0	Z-87.8	$Z-90.0 \text{ Part Dim.} + 3.0 \text{ Radius} - .8 \text{ Tool Radius}$
P30	X 15.2 12.2	Z-90.0 K0	$X 16.0 \text{ Part Radius} - .8 \text{ Tool Radius}$ $Z-90.0 \text{ Part Dim.}$
P31	X 18.5	Z-90.0	$X 16.0 \text{ Part Radius} + 2.5 \text{ Clearance}$
P32	TOOL	CHANGE	POSITION



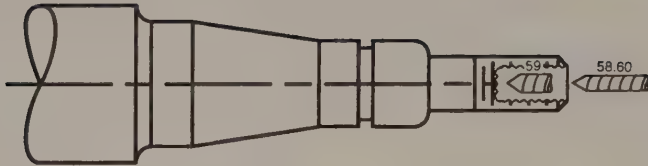
CUT GROOVE

POINT CODE	X COORDINATE	Z COORDINATE	CALCULATION
P33	X 12.5	Z-43.0	$X 10.0 \text{ Part Radius} + 2.5 \text{ Clearance}$ $Z-40.0 \text{ Part Dim.} - 3.0 \text{ Groove Width}$
P34	X 6.5	Z-43.0	$X 6.5 \text{ Groove Radius}$
P35	X 12.5	Z-43.0	$X 10.0 \text{ Part Radius} + 2.5 \text{ Clearance}$
P36	TOOL	CHANGE	POSITION



THREAD CUTTING

POINT CODE	X COORDINATE	Z COORDINATE	CALCULATION
P37	X 9.5	Z 2.5	X 7.0 Part Radius + 2.5 Clearance Z 0 Finish Datum + 2.5 Clearance
P38	X 6.1	Z 1.98	X 7.0 Part Radius - .9 Thread In Feed Z 2.5 Clearance - .52 Thread In Feed K (Thread Pitch)
P39	X 6.1	Z-19	Z-19 Thread Length Dim.
P40	X 9.5	Z-19	X 7.0 Part Radius + 2.5 Clearance
P42	X 5.4	Z 1.58	X 7.0 Part Radius - 1.6 Thread In Feed Z 2.5 Clearance - .92 Thread In Feed
P46	X 4.9	Z 1.29	X 7.0 Part Radius - 2.1 Thread In Feed Z 2.5 Clearance - 1.21 Thread In Feed
P50	X 4.546	Z 1.09	X 7.0 Part Radius - 2.454 Thread In Feed Z 2.5 Clearance - 1.41 Thread In Feed
	TOOL	CHANGE	POSITION



DRILL 6MM DIAMETER HOLE

POINT

CODE	X COORDINATE	Z COORDINATE	CALCULATION
P58	X 0	Z 2.5	X 0 Datum Z 0 Datum + 2.5 Clearance
P59	X 0	Z-13.8	Z-12 Hole Depth - 1.8 Drill Point .3 X Dia. Drill Point Allowance
P60	X 0	Z 2.5	X 0 Datum Z 0 Datum + 2.5 Clearance
P61	TOOL	CHANGE	POSITION

Lathe—Metric Program

```

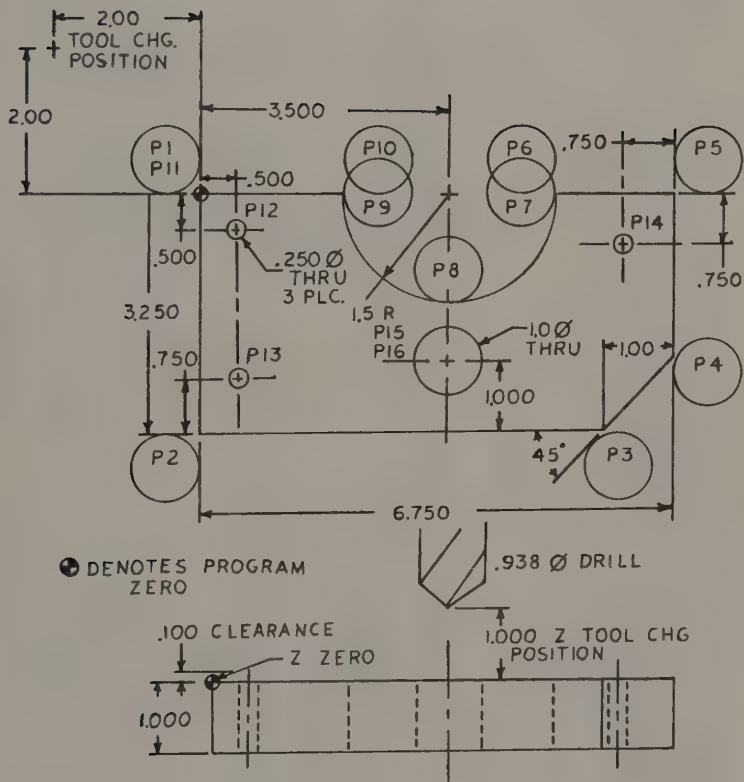
%      Rewind stop code
N0010 G71      Metric programming
N0020 G97      rpm spindle speed programming
N0030 G95      mmpr feedrate programming
N0040 G90      Absolute coordinate programming
N0050 G00 T00   Clear tool offsets
N0060 T0600     Index rear turret to empty station
N0070 T0101 S960 M03   Index tool 1 to cutting position (rough face and turn),
                        start spindle
N0080 G92 X114.0 Z166.0   Preset X & Z axes
N0090 G00 X18.5 Z.75 M08   Position tool to rough face, turn coolant on (P1)
N0100 G01 X-1.3 F.25      Rough face part leaving 0.5 stock (P2)
N0110 G00 Z3.25          Retract from face (P3)
N0120 X13.6             Position X for first rough turn (P4)
N0130 G01 Z-87.8 F.25     Feed for first rough turn (P5)
N0140 G00 X14.8 Z-86.6     Retract X & Z axes (P6)
N0150 Z1.25 S1132         Rapid to face of part (P7)
N0160 X10.6             Position X for second rough turn (P8)
N0170 G01 Z-50.72 F.25     Feed for second rough turn (P9)
N0180 G00 X11.8 Z-49.52    Retract X & Z axes (P10)
N0190 Z1.25 S2000         Rapid to face of part (P11)
N0200 X7.4             Position X for semifinish pass (P12)
N0210 G01 Z-27.78         Feed tangent to convex radius (P13)
N0220 S1538             Change rpm
N0230 G03 X10.4 Z-31.8 I1.2 K4.02   Contour convex radius (P14)
N0240 G01 Z-50.73         Feed tangent to taper (P15)
N0250 S1200             Change rpm
N0260 X13.4 Z-78.73       Feed up taper (P16)
N0270 Z-87.8            Feed tangent to concave radius (P17)
N0280 S1010             Change rpm
N0290 G02 X15.2 Z-89.6 I1.8 K0       Contour concave radius (P18)
N0300 G01 X18.5          Feed off part (P19)
N0310 G00 X114.0 Z166.0 T0000   Rapid to tool change position—cancelling
                                tool offset (P20)
N0320 G92 X0 Z0          Cancel axis presets
N0330 G00 T0202         Index finish facing and turning tool into cutting position
N0340 G90 S2000 M03      Set finishing rpm
N0350 G92 X114.0 Z166.0   Axis presets
N0360 G00 X9.9 Z0        Rapid to start finish face (P21)
N0370 G01 X-1.3 F.12     Finish face part (P22)
N0380 G00 X2.53 Z2.5     Retract in Z and position for chamfer (P23)
N0390 G01 X7.0 Z-1.97    Turn chamfer (P24)
N0400 Z-25.39           Finish turn to convex radius (P25)
N0410 S1600             Change rpm
N0420 G03 X10.0 Z-31.8 I.8 K3.71   Contour convex radius (P26)
N0430 G01 Z-50.75       Finish turn tangent to taper (P27)
N0440 S1200             Change rpm
N0450 X13.0 Z-78.75     Finish turn taper (P28)
N0460 Z-87.8           Finish turn tangent to concave radius (P29)
N0470 S1010             Change rpm

```

N0480 G02 X15.2 Z-90.0 I2.2 K0 *Contour concave radius* (P30)*
 N0490 G01 X18.5 *Feed away from part (P31)*
 N0500 G00 X114.0 Z166.0 T0000 M09 *Rapid to tool change position cancelling tool offsets and turning off coolant (P32)*
 N0510 G92 X0 Z0 *Cancel axis presets*
 N0520 G00 T0303 *Index grooving tool into cutting position*
 N0530 G90 S1000 M03 *Start spindle*
 N0540 G92 X114.0 Z166.0 *Preset axes*
 N0550 G00 X12.5 Z-43.0 M08 *Position in X & Z to cut groove (P33)*
 N0560 G01 X6.5 F.12 *Cut groove to depth (P34)*
 N0570 G04 F05 *Dwell for 5 sec*
 N0580 G00 X12.5 M09 *Retract from groove turning coolant off (P35)*
 N0590 G00 X114.0 Z166.0 T0000 *Rapid to tool change position cancelling tool offsets (P36)*
 N0600 G92 X0 Z0 *Cancel axis presets*
 N0610 G00 T0404 *Index threading tool into cutting position*
 N0620 G90 S1000 M03 *Start spindle*
 N0630 G92 X114.0 Z166.0 *Preset axes*
 N0640 G00 X9.5 Z2.5 M08 *Position in X & Z to cut thread (P37)*
 N0650 X6.1 Z1.98 *First thread in feed (P38)*
 N0660 G33 Z-19.0 K2.0 *Cut first thread pass (P39)*
 N0670 G00 X9.5 *Retract X (P40)*
 N0680 Z2.5 *Rapid to end of part (P41)*
 N0690 X5.4 Z1.58 *Thread in feed for second pass (P42)*
 N0700 G33 Z-19.0 K2.0 *Cut second pass (P43)*
 N0710 G00 X9.5 *Retract X (P44)*
 N0720 Z2.5 *Rapid to end of part (P45)*
 N0730 X4.9 Z1.29 *Thread in feed for third pass (P46)*
 N0740 G33 Z-19.0 K2.0 *Cut third pass (P47)*
 N0750 G00 X9.5 *Retract X (P48)*
 N0760 Z2.5 *Rapid to end of part (P49)*
 N0770 X4.546 Z1.09 *Thread in feed for finish pass (P50)*
 N0780 G33 Z-19.0 K2.0 *Cut finish pass (P51)*
 N0790 G00 X9.5 *Retract X (P52)*
 N0800 Z2.5 *Rapid to end of part (P53)*
 N0810 X4.546 Z1.09 *Thread in feed for tool pressure pass (P54)*
 N0820 G33 Z-19.0 K2.0 *Cut tool pressure pass (P55)*
 N0830 G00 X9.5 M09 *Retract X (P56)*
 N0840 X114.0 Z166.0 T0000 *Rapid to tool change cancelling tool offsets (P57)*
 N0850 G92 X0 Z0 *Cancel axis preset*
 N0860 G00 T0707 *Index 6.0 mm drill into cutting position*
 N0870 G90 S2000 M03 *Start spindle*
 N0880 G92 X-114.0 Z52.0 *Preset axes*
 N0890 G00 X0 Z2.5 M08 *Position to drill (P58)*
 N0900 G01 Z-13.8 F.15 *Drill to depth (P59)*
 N0910 G00 Z2.5 M09 *Retract drill (P60)*
 N0920 X-114.0 Z166.0 T0000 *Rapid to tool change cancelling tool offsets (P61)*
 N0930 G92 X0 Z0 *Cancel axis presets*
 N0940 M02 *End of program*

Example 3: Milling, Inch

MILL EXAMPLE INCH



MACHINING CENTER - INCH PROGRAMMING

TOOL SHEET

PART NO. NC-2118906		MATERIAL 1018 H.S.		OPERATION 20		PROGRAMMED BY T. CRANDELL	
PART NAME MILL EXAMPLE		B/P CHANGE DATE		DATE		SHEET 1 OF 1	
TOOL		TOOL DIAMETER		TOOL LENGTH		SPEED	
SEQ. NUMBER	DESCRIPTION	PROG.	ACT.	#	PROG.	R P M	FEED
01	1.0 DIA. FOUR FLUTE ENDMILL #56779 HSS National Twist HOLDER # QC40-EM100-338 Kennametal	1.000			5.500 PROJECTION 2.00 MIN.	400 RPM	11.2 IPM
02	.250 DIA. DRILL HSS #49016 National Twist HOLDER # QC40-DA300-163 Kennametal COLLECT # 300DA-0250	.250			4.500 PROJECTION 2.5 MIN.	1600 RPM	9.6 IPM
03	15/16 DIA. DRILL HSS #10472 National Twist HOLDER #2-60-209-303 Kennametal	.938			7.000 PROJECTION 3.5 MIN.	425 RPM	8.5 IPM
04	1.000 DIA. BORING TOOL CARB. BORING BAR # 2-80-076-026 Kennametal Insert #E32ACP HOLDER # QC40-TG150-450 Kennametal	1.000			6.000 PROJECTION 2.5 MIN.	1200 RPM	7.2 IPM

INCH
MACHINING CENTER
COORDINATE SHEET

COMPANY FERRIS STATE UNIVERSITY

DATE

PART NO. NC-2118906		PART NAME MILL EXAMPLE		OPERATION NO. 20			BY: BRANDELL SHEET 1 OF 5		
POINT CODE	X ABSOLUTE COORDINATE	Y ABSOLUTE COORDINATE	TABLE DEG. ARS.	TOOL DESCRIPTION	T SETTING LENGTH	D DEPTH BELOW WORK SUR.	S WORK SUR.	Z CLEAR POS.	Z DEPTH POS.
								TOOL - PRESENT + TOOL CHG. CLEAR- ANCE	WORK SUR. + CLEAR- ANCE FEATURE DP. - TOOL POINT
TOOL CHG.	-2.000	+2.000		TO1 1.0 END MILL	7.000	0	0	1.000	1.000
	X 0 Datum - 2.000 Cutter Center Position Y 0 Datum + 2.000 Cutter Center Position				Tool Chg. 7.000 - 7.000 + 1.000 = 1.000 Clear Pos. 0 + 1.000 = 1.00				
P1	-.500	+.500			5.500	1.100	0	2.500	.100
	X 0 Datum - .500 Cutter Radius Y 0 Datum + .500 Cutter Radius				Tool Chg. 7.000 - 5.500 + 1.000 = 2.500 Clear Pos. 0 + .100 = .100 Z Depth 0-1.0 Part Thickness-.100 Clear=-1.100				
P2	-.500	-3.750							
	Y 0 Datum - 3.750 Part Width - .500 Cutter Radius								
P3	5.9413	-3.750							
	67.5° 				X 0 + 6.75 Part Length - 1.00 Angle Length + COTAN 67.5° * .5 Cutter Center = 5.75 + .2071 = 5.9571				
P4	7.250	-2.4571							
	X 0 Datum + 6.75 Part Length + .500 Cutter Radius = 7.250 Y 0 Datum - 1.25 Part Width + 1.00 Angle Width Length - COTAN 67.5° * .5 Cutter Center = -2.25 - .2071 = -2.4571				Continued				

INCH
MACHINING CENTER
COORDINATE SHEET

COMPANY FERRIS STATE UNIVERSITY

DATE

PART NO. NC-2118906		PART NAME MILL EXAMPLE		OPERATION NO. 20		SHEET 2 OF 5					
POINT CODE	X ABSOLUTE COORDINATE	Y ABSOLUTE COORDINATE	TABLE DEG. ANG.	TOOL DESCRIPTION		T SETTING LENGTH	D DEPTH BELOW WORK SUR.	S WORK SUR.	Z TOOL CHG.	Z CLEAR POS.	Z DEPTH POS.
				Note: The longest tool (.938 dia. drill 7.00 gage length) is used for setting the Z tool chg.							
			67.5°								
P5	7.250	.500									
X 0 Datum + 6.750 Part Length + .500 Cutter Radius Y 0 Datum + .500 Cutter Radius											
P6	4.500	.500									
X 0 Datum + 3.500 Part Radius Center + 1.500 Part Radius - .500 Cutter Radius											
P7	4.500	0									
Y 0 Datum											
P8	3.500	-1.000									
I 1.0 J0											
X 0 Datum + 3.500 Part Radius Center Y 0 - 1.500 Part Radius + .500 Cutter Radius											
P9	2.500	0									
I 0 J1.0											
X 0 Datum + 3.500 Part Radius Center - 1.500 Part Radius + .500 Tool Radius Y 0 Datum J 1.500 Part Radius - .500 Cutter Radius											

TECH
MACHINING CENTER
COORDINATE SHEET

COMPANY FERRIS STATE UNIVERSITY

DATE

PART NO. NC-2118906		PART NAME MILL EXAMPLE		OPERATION NO. 20			BY: RANDELL SHEET 3 OF 5					
POINT CODE	X ABSOLUTE COORDINATE	Y ABSOLUTE COORDINATE	TABLE DEG. ABS.	TOOL DESCRIPTION	T SETTING LENGTH	D DEPTH BELOW WORK SUR.	S WORK SUR.	Z TOOL CHG.	Z CLEAR POS.	Z DEPTH POS.	WORK SUR.	WORK SUR.
P10	2.500	.500										
Y O Datum + .500 Cutter Radius												
P11	-.500	.500										
X O Datum - .500 Cutter Radius Y O Datum + .500 Cutter Radius												
TOOL CHG.	-2.000	2.000		T02 .250 DIA. DRILL	5.500 TL.#1	0	0	2.500				
X O Datum - 2.000 Tool Change Position Y O Datum + 2.000 Tool Change Position Z 7.000 Set TL. - 5.500 Present TL.#1 + 1.000 Clearance = 2.500												
P12	.500	-.500			4.500 TL.#2	1.175	0	3.500	.100	-1.175		
X O Datum + .500 Hole Center Dim. Y O Datum - .500 Hole Center Dim. Tool Chg. 7.000 - 4.500 + 1.000 = 3.500 Clear Pos. 0 + .100 = .100 Z Depth 0 - 1.00 - .175 = -1.175												
P13	.500	-2.500										
Y O Datum - 3.250 Part Width + .750 Hole Center Dim.												

[illegible]

Inch Machining Center Example Program

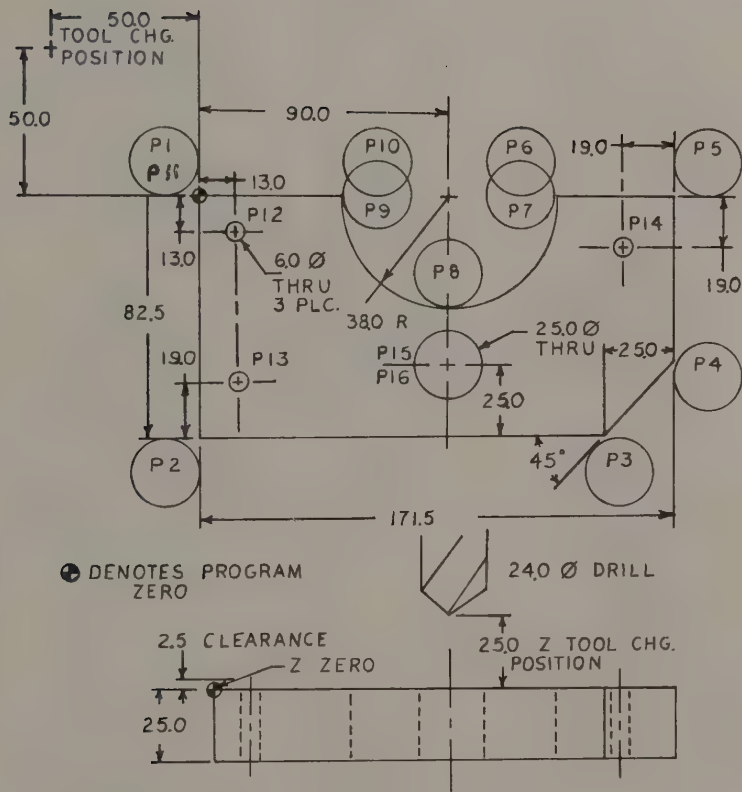
```

N0010 G90      Absolute programming
N0020 G70      Inch programming
N0030 G94      Inch per minute feedrate
N0040 G17      X-Y circular interpolatiòn plane
N0050 G40 T00   Cancel cutter diameter compensation and tool length com-
                pensation
N0060 G80 Z1.0   Retract Z axis
N0070 G00 X-2.0 Y2.0 T01 M06   Tool change "1.0 end mill"
N0080 X-.5 Y.5 Z.1 S400 M03   Rapid to position 1, start spindle
N0090 Z-1.1 M08   Rapid to Z depth, turn coolant on
N0100 G01 Y-3.75 F11.2   Feed to position 2
N0110 X5.9413   Feed to position 3
N0120 X7.25 Y-2.4571   Feed to position 4
N0130 Y.5   Feed to position 5
N0140 X4.5   Feed to position 6
N0150 Y0   Feed to position 7
N0160 G02 X3.5 Y-1.0 I1.0 J0   Circular interpolate to position 8
N0170 X2.5 Y0 I0 J1.0   Circular interpolate to position 9
N0180 G01 Y.5   Feed to position 10
N0190 X-.5   Feed to position 11
N0200 G00 Z.1 M09   Clear part
N0210 G80 X-2.0 Y2.0 Z2.5 T02 M06   Tool change ".250 dia. drill"
N0220 X.5 Y-.5 Z.1 S1600 M03   Rapid to position 12 turn, spindle on
N0230 G81 Z-1.175 R.1 F9.6 M08   Drill position 12, turn coolant on
N0240 Y-2.5   Drill position 13
N0250 X6.0 Y-.75 M09   Drill position 14, turn coolant off
N0260 G80 X-2.0 Y2.0 Z3.5 T03 M06   Tool change ".938 dia drill"
N0270 X3.5 Y-2.25 Z.1 S425 M03   Rapid to position 15, turn spindle on
N0280 G81 Z-1.3814 R.1 F8.5 M08   Drill position 15, turn coolant on
N0290 M09   Turn coolant off
N0300 G80 X-2.0 Y2.0 Z1.0 T04 M06   Tool change "1.0 dia. boring bar"
N0310 X3.5 Y-2.25 Z.1 S1200 M03   Rapid to position 16, turn spindle on
N0320 G81 Z-1.1 R.1 F7.2 M08   Bore position 16, turn coolant on
N0330 M09   Turn coolant off
N0340 G80 X-2.0 Y2.0 Z2.0 M02   Rapid to tool change position and end pro-
                                gram

```


Example 4: Milling, Metric

MILL EXAMPLE METRIC



MACHINING CENTER - METRIC PROGRAMMING

TOOL SHEET

PART NO. NC-2118906		MATERIAL 1018 M.S.		OPERATION 20		PROGRAMMED BY T. CRANDELL	
PART NAME MILL EXAMPLE		B/P CHANGE DATE		DATE		SHEET 1 OF 1	
TOOL	TOOL	TOOL DIAMETER		TOOL LENGTH		SPEED	
SEQ. NUMBER	DESCRIPTION	PROG.	ACT.	#	PROG. ACT.	R.P.M.	FEED
01	24.0 DIA. FOUR FLUTE ENDMILL # D3242-EDF24449 HSS Vern Wheeler Co. HOLDER # QC40-EM100-338 Kennametal	24.0MM			PROJECTION 50.0MM MIN.	400 RPM	284 MM/ MIN.
02	6.0 DIA. DRILL HSS #12695 National Twist HOLDER # QC40-DA300-163 Kennametal COLLECT #300DA-0250	6.0MM			114.0MM PROJECTION 64MM MIN.	1600 RPM	244 MM/ MIN.
03	24.0 DIA. DRILL HSS # HOLDER #	24.0MM			178.0MM PROJECTION 90MM MIN.	425 RPM	216 MM/ MIN.
04	25.0 DIA BORING TOOL CARB. BORING BAR # 2-80-076-026 Kennametal INSERT #E32ACP HOLDER # QC40-TG150-450 Kennametal	25.0MM			152.0MM PROJECTION 64MM MIN.	1200 RPM	183 MM/ MIN.

METRIC
MACHINING CENTER
COORDINATE SHEET

COMPANY FERRIS STATE UNIVERSITY

DAVID

PART NO. RC-2118906			PART NAME HILL EXAMPLE			OPERATION NO. 20			BY CRANDALL SHEET 1 OF 5											
POINT CODE	X ABSOLUTE COORDINATE	Y ABSOLUTE COORDINATE	TABLE DEG.	TOOL DESCRIPTION	T SETTING LENGTH	D DEPTH BELOW WORK SUR.	S WORK SUR.	Z TOOL CHG.	Z CLEAR POS.	Z CLEAR POS.	DEPTH POS.	WORK SUR.	THICKNESS ANCE	PRESENT TOOLS + CLEARANCE	FEATURE DP. - TOOL POINT					
P1	-50.0	+50.0							TOI 25.0 ENDMILL	178.0 SET TOOL	0	0		25.0						
Tool Chg. 178.0 - 178.0 + 25.0 = 25.0 Clear Pos. 0 + 25.0 = 25.0																				
P2	-12.5	+12.5								140.0	27.5	0		63.0						
Tool Chg. 178 - 140 + 25.0 = 63.0 Clear Pos. 0 + 2.5 = 2.5 Z Depth 0 - 25.0 Part Thickness -2.5 Clear=27.5																				
Y O Datum - 82.5 Part Width - 12.5 Cutter Radius																				
P3	151.68	-95.0																		
X O ± 171.5 Part Length - 25.0 Angle Length + COTAN 67.5° * 12.5 Cutter Center = 166.5 + 5.18 = 151.68																				
P4	184.0	-62.68																		
X O Datum + 171.5 Part Length + 12.5 Cutter Radius = 184.0 Y O Datum - 82.5 Part Width + 25.0 Angle Length - COTAN 67.5° * 12.5 Cutter Center = -57.5 - 5.18 = -62.68																				

METRIC
MACHINING CENTER
COORDINATE SHEET

COMPANY FERRIS STATE UNIVERSITY

DATE

PART NO. NC-2118906		PART NAME MILL EXAMPLE		OPERATION NO. 20			SYCRANDELL SHEET 2 OF 5		
POINT CODE	X ABSOLUTE COORDINATE	Y ABSOLUTE COORDINATE	TABLE DEG. AMS.	TOOL DESCRIPTION	Y SETTING LENGTH	D DEPTH BELOW WORK SUR.	S WORK SUR.	Z CLEAR POS.	Z DEPTH POS.
Note: The longest tool (24.0 dia. drill 178.0 gage length) is used for setting the 2 tool chg. 67.5°									
P5	184.0	12.5							
X 0 Datum + 171.5 Part Length + 12.5 Cutter Radius Y 0 Datum + 12.5 Cutter Radius									
P6	115.5	12.5							
X 0 Datum + 90.0 Part Radius Center + 38.0 Part Radius - 12.5 Cutter Radius									
P7	115.5	0							
Y 0 Datum									
P8	90.0	-25.5		I 25.5 J0					
X 0 Datum + 90.0 Part Radius Center Y 0 Datum - 38.0 Part Radius + 12.5 Cutter Radius I 38.0 Part Radius - 12.5 Cutter Radius									
P9	64.5	0		I0 J 25.5					
X 0 Datum + 90.0 Part Radius Center - 38.0 Part Radius + 12.5 Cutter Radius Y 0 Datum J 38.0 Part Radius - 12.5 Cutter Radius									

METRIC
MACHINING CENTER
COORDINATE SHEET

COMPANY FERRIS STATE UNIVERSITY

DATE

PART NO. NC-2118906			PART NAME MILL EXAMPLE		OPERATION NO. 20		BY: RANDELL		SHEET 3 OF 5	
POINT CODE	X ABSOLUTE COORDINATE	Y ABSOLUTE COORDINATE	TABLE DEC. ABS.	TOOL DESCRIPTION	T SETTING LENGTH	D DEPTH BELLOW WORK SUR.	S WORK SUR.	Z TOOL CHG.	Z CLEAR POS.	Z DEPTH POS.
								TOOL - PRESENT + TOOL CHG. CLEAR- ANCE	SET WORK SUR. + CLEAR- ANCE	WORK SUR. - FEATURE DP. - TOOL POINT
P10	64.5	12.5								
Y 0 Datum + 12.5 Cutter Radius										
P11	-12.5	12.5								
X 0 Datum - 12.5 Cutter Radius Y 0 Datum + 12.5 Cutter Radius										
TOOL CHG.	-50.0	50.0		T02 6.0 DIA. DRILL	140.0 TL. #1	0	0	63.0		
X 0 Datum - 50.0 Tool Chg. Position Y 0 Datum + 50.0 Tool Chg. Position Z 178.0 Set TL. - 140.0 Present TL. #1 + 25.0 Clearance = 63.0										
P12	13.0	-13.0			114.0 TL. #2	29.3	0	89.0	2.5	-29.3
X 0 Datum + 13.0 Hole Center Dim. Y 0 Datum + 13.0 Hole Center Dim. Tool Chg. 178.0 - 114.0 + 25.0 = 89.0 Clear Pos. 0 + 2.5 = 2.5 Z Depth 0 - 25 - 4.3 = -29.3										
P13	13.0	-63.5								
Y 0 Datum - 82.5 Part Width + 19.0 Hole Center Dim.										

METRIC
MACHINING CENTER
COORDINATE SHEET

COMPANY FERRIS STATE UNIVERSITY

DATE

PART NO. NC-2118906		PART NAME MILL EXAMPLE		OPERATION NO. 20		BY CARD DEL		SHEET 4 OF 5		
POINT CODE	X ABSOLUTE COORDINATE	Y ABSOLUTE COORDINATE	TABLE DEG. ANG.	TOOL DESCRIPTION	T SETTING LENGTH	D DEPTH FROM WORK SUR.	S WORK SUR.	Z TOOL CHG.	Z CLEAR POS.	Z DEPTH POS.
P14	152.5	-19.0								
X 0 Datum + 171.5 Part Length - 19.0 Hole Center Dim. Y 0 Datum - 19.0 Hole Center Dim.										
TOOL CHG.	-50.0	50.0		T03 24.0 DIA. DRILL	114.0 TL.#2	0	0	89.0		
X 0 Datum - 50.0 Tool Chg. Position Y 0 Datum + 50.0 Tool Chg. Position Z 178.0 Set TL. - 114.0 Present TL. #2 + 25.0 Clearance = 89.0										
P15	90.0	-57.5			178.0 TL.#3	34.7	0	25.0	2.5	-34.7
X 0 Datum + 90.0 Hole Center Dim. Y 0 Datum - 82.5 Part Width + 25.0 Hole Center Dim. Tool Chg. 178.0 - 178.0 + 25.0 = 25.0 Clear Pos. 0 + 2.5 = 2.5 Z Depth 0 - 25.0 - 7.2 - 2.5 = -34.7										
TOOL CHG.	-50.0	50.0		T04 25.0 DIA. BORING BAR	178.0 TL.#3	0	0	25.0		
X 0 Datum - 50.0 Tool Change Position Y 0 Datum + 50.0 Tool Change Position Z 178.0 Set TL. - 178.0 Present TL. #3 + 25.0 Clearance = 25.0										
P16	90.0	-57.5			152.0 TL.#4	27.5	0	51.0	2.5	-27.5
X 0 Datum + 90.0 Hole Center Dim. Y 0 Datum - 82.5 Part Width + 25.0 Hole Center Dim. Tool Chg. 178.0 - 152.0 + 25.0 = 51.0 Clear Pos. 0 + 2.5 = 2.5 Z Depth 0 - 25.0 - 2.5 = -27.5										

**METRIC
MACHINING CENTER
COORDINATE SHEET**

COMPANY FERRIS STATE UNIVERSITY

DATE _____

[illegible]

Metric Machining Center Example Program

```

N0010 G90      Absolute programming
N0020 G71      Metric programming
N0030 G94      Millimeter per minute feedrate
N0040 G17      X-Y circular interpolation plane
N0050 G40 T00   Cancel cutter diameter compensation and tool length com-
                pensation
N0060 G80 Z25.0 Retract Z axis
N0070 G00 X-50.0 Y50.0 T01 M06   Tool change "25.0 endmill"
N0080 X-12.5 Y12.5 Z2.5 S400 M03   Rapid to position 1, start spindle
N0090 Z-27.5 M08   Rapid to Z depth turn coolant on
N0100 G01 Y-95.0 F284   Feed to position 2
N0110 X151.68   Feed to position 3
N0120 X184.0 Y-62.68   Feed to position 4
N0130 Y12.5     Feed to position 5
N0140 X115.5     Feed to position 6
N0150 Y0        Feed to position 7
N0160 G02 X90.0 Y-25.5 I25.5 J0   Circular interpolate to position 8
N0170 X64.5 Y0 I0 J25.5   Circular interpolate to position 9
N0180 G01 Y12.5   Feed to position 10
N0190 X-12.5     Feed to position 11
N0200 G00 Z2.5 M09   Clear part
N0210 G80 X-50.0 Y50.0 Z63.0 T02 M06   Tool change "6.0 dia. drill"
N0220 X13.0 Y-13.0 Z2.5 S1600 M03   Rapid to position 12, turn spindle on
N0230 G81 Z-29.3 R2.5 F244 M08   Drill position 12, turn coolant on
N0240 Y-63.5     Drill position 13
N0250 X152.0 Y-19.0   Drill position 14, turn coolant off
N0260 G80 X-50.0 Y50.0 Z89.0 T03 M06   Tool change "24.0 dia drill"
N0270 X90.0 Y-57.5 Z2.5 S425 M03   Rapid to position 15, turn spindle on
N0280 G81 Z-34.7 R2.5 F216 M08   Drill position 15, turn coolant on
N0290 M09        Turn coolant off
N0300 G80 X-50.0 Y50.0 Z25.0 T04 M06   Tool change, "25.0 dia boring bar"
N0310 X90.0 Y-57.5 Z2.5 S1200 M03   Rapid to position 16, turn spindle on
N0320 G81 Z-27.5 R2.5 F183 M08   Bore position 16, turn coolant on
N0330 M09        Turn coolant off
N0340 G80 X-50.0 Y50.0 Z51.0 M02   Rapid to tool change position and end
                                   program

```

CONVERSATIONAL PART PROGRAMMING

Conversational part programming requires the programmer to respond to a set of questions that is a built-in feature of the machine control system and displayed on the CRT screen of the control unit. As each response is made, further options are presented and responses made until that particular group, or "block" of related data, is complete. The programmer then moves on to the next block.

For example, assume that, as part of a milling program, a relative tool movement of -39.786 mm (-1.568 in.) is required in a certain direction identified as the X axis. An appropriate feedrate has already been programmed.

With conversational programming the programmer will establish the appropriate operating mode, that is, linear motion at a controlled feedrate, by making a selection from a list of alternatives pushing the appropriate button. That selection having been made, the next prompt will ask for the dimensional value of the intended move in the *X* axis to be keyed in by the operator.

If listed an example of the data for the above move could read as follows: Note different controls will use varying formats.

N260 MILL or LIN X-39.786

where the entry MILL or LIN (linear) indicates the type of slide movement required and N260 is the data block number.

Similarly, consider a program entry to achieve combined slide movements that will result in a cutter path passing through an arc of 90° and having a radius 8 mm (0.3 in.).

First, the appropriate operating mode will be selected, such as CIRC—an abbreviation for circular interpolation. The prompts that follow will ask for the target position in the respective axes followed by the value of the radius and the direction of movement, whether clockwise or counterclockwise.

A listed program block containing these data would read

N350 CIRC X43.765 Z-75.000 R8 CW

The conversational concept can be extended to include machining requirements other than slide movement control.

Consider the turning of a bar of metal on a turning center. Before any thought can be given to slide movements, the basic metal-cutting data would have to be ascertained. For example, the correct spindle speed and feedrate are of vital importance. The spindle speed is affected by the work diameter and the cutting speed. The cutting speed is related to the material being machined. The feed rate would depend on the depth of cut, tool type, and surface finish required.

From this it can be seen that the necessary data to machine the metal successfully can be related to four factors:

- (a) the material being cut;
- (b) the material diameter;
- (c) the surface finish required;
- (d) the tool type.

Some computerized controls can be programmed to select the appropriate spindle speed and feedrate from an input of information relating to these factors. To assist the input of information there will be a material file and surface roughness file within the computer memory, as shown in Figure 8.66. Cutting tool types will also be numerically identified. This type of input could be found on a machine control unit, but is more likely to be encountered on offline programming systems.

MATERIAL STOCK FILE	
CODE	MATERIAL
1	MILD STEEL
2	MED. CARBON STEEL
3	STAINLESS STEEL
4	CAST IRON
5	ALUMINUM

SURFACE ROUGHNESS FILE	
CODE	Ra (μ in.)
1	100 (250)
2	50 (125)
3	25 (63)
4	12.5 (32)
5	6.3 (16)

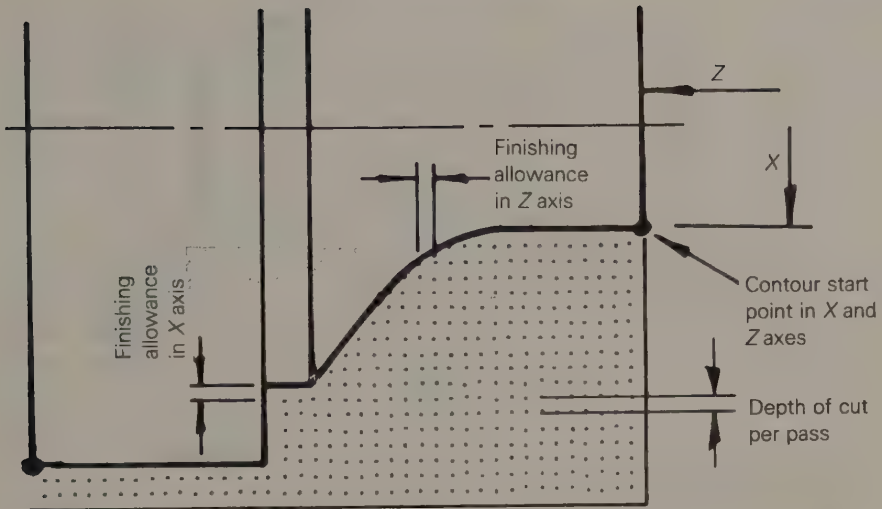
Figure 8.66 *Material file and surface roughness file.*

A simple question-and-answer routine will extract from the computer memory all the necessary data to give the correct cutting conditions, making calculations or judgments on the part of the programmer quite unnecessary. An example of a question-and-answer routine is as follows:

Prompt	Response
Material?	5
Material diameter?	50
Surface code?	4
Tool number?	8

There is no standard conversational part programming language. The systems are very individualistic. It should also be noted that while conversational MDI programs can be prepared away from the machine and then instantly entered into the control, usually by the use of magnetic tape, floppy disk, or direct numerical control, they are commonly entered by the machine setup person/operator pressing appropriate buttons as described previously. Unless the machine control unit is of the less common type that permits a second program to be entered while the first is being activated, shop floor data entry involves the machine being nonproductive while the program is being entered, and this does have its drawbacks. On the other hand, the technique is favored by some companies since total control of the machining operation by workshop personnel (as opposed to programs being prepared away from the shop floor and then subsequently passed on to them), makes use of valuable practical skills and, equally important, has the effect of improving job satisfaction.

Conversational controls are also unique in their programming in that they have many specialized routines in order to make part material removal and repetitive operations easier. Figure 8.67 shows a very useful stock removal



Further data required:

- (1) Profile data (added at end of program);
- (2) Profile data start block number;
- (3) Profile data end block number.

A final profile trace is optional.

Figure 8.67 Stock removal to a defined contour cycle. (Refer to your programming manual for specific examples.)

cycle. From one data block, plus blocks defining the component profile, the controller will automatically determine the number of passes necessary to remove the excess material, constantly varying the length of travel in the Z axis if need be, and finally taking a finishing cut to reduce the component profile precisely to size. The definition of the component profile in program data is added at the end of the program and is automatically activated via the stock removal cycle call. The profile definition can be achieved by the inclusion of appropriate minor cycles such as those described above.

This stock removal cycle is for application along the Z axis. Similar cycles cater for stock removal along the face or X axis of a workpiece.

Figures 8.68 and 8.69 illustrate cycles that may be used for reducing the diameter at any position along the length of a part and for grooving. Figure 8.70 illustrates a screw-cutting cycle, an essential feature of any control system devoted to turning. This particular version of a screw-cutting cycle is particularly easy to use. From just one block of data the control automatically de-

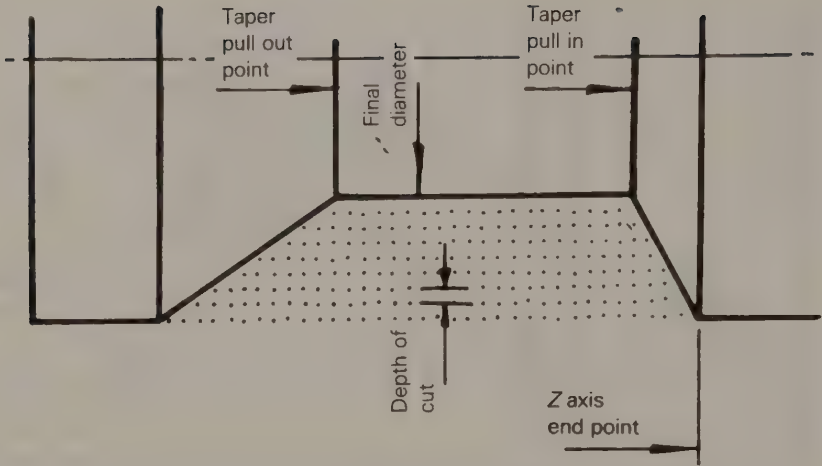


Figure 8.68 Stock removal cycle with optional tapered entry and exit.

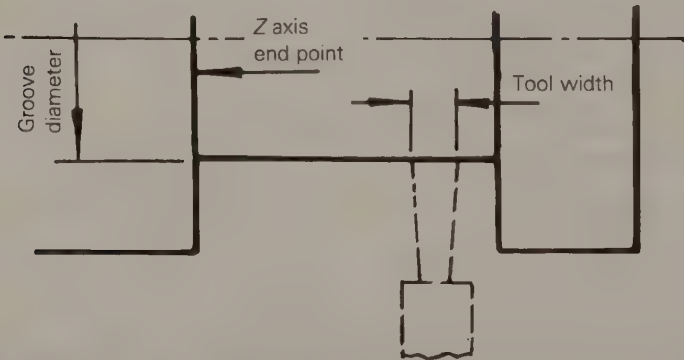
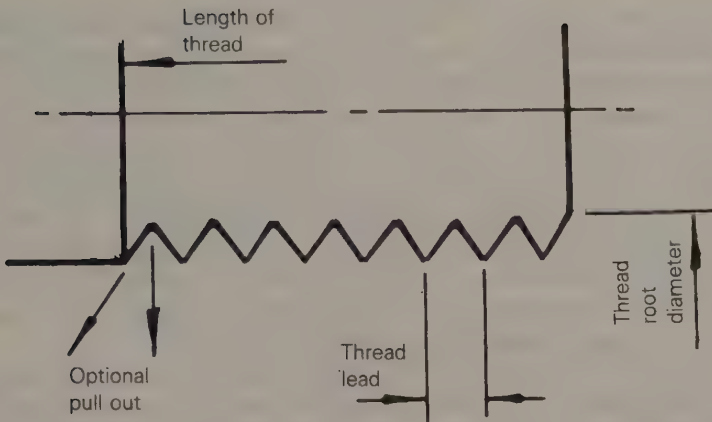


Figure 8.69 Stock removal or grooving cycle.

terminates the number of passes necessary to achieve the required thread depth. In the block of data shown G84 is the cycle code, X specifies the thread root diameter, Z specifies the thread length, the P_1 value is the depth of the first pass, and P_2 is the lead. The control automatically effects a progressive reduction in the depth of cut of each pass that results in improved surface finish and prolonged tool life.

On less sophisticated screw-cutting cycles the X diameter for each pass along the thread length has to be predetermined by the part programmer, and each pass is programmed in a separate data block.

The range of special cycles used in milling machine control systems is equally helpful to the part programmer.



Data example: G84X8.168Z-20P₁ IP₂ I.5

Figure 8.70 Automatic threading cycle (millimeter example).

Fairly common is the provision of a face milling cycle such as that illustrated in Figure 8.71, where a data input specifies the dimensions of the face to be milled. From this information the control unit will determine the number of passes required while taking into consideration a stated cutter overlap that will ensure the face is evenly machined.

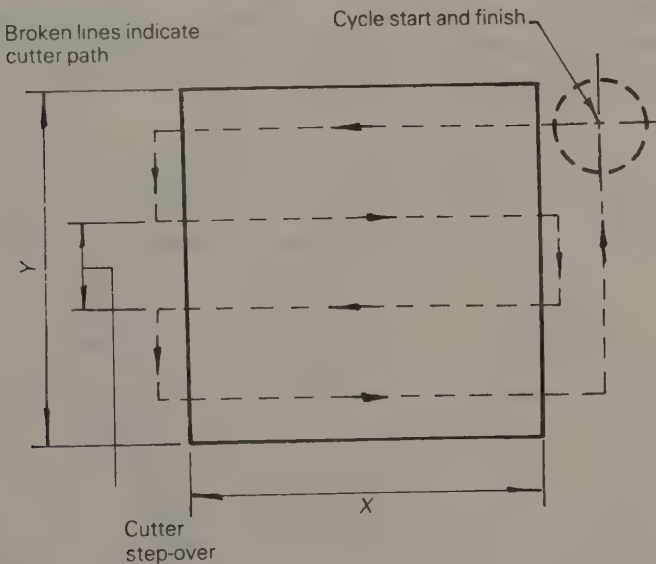


Figure 8.71 Face milling cycle.

Figure 8.72 illustrates a slot milling cycle. Here again the overall dimensions are programmed. The first pass made by the cutting tool goes through the center and then returns to the start. Further passes are made until the correct depth is achieved, the number of passes necessary being determined from the programmed movement to be made in the Z axis before each cut commences. When the correct depth is reached, the cutter path will be a series of loops increasing in size with each pass. As with the face milling cycle, the control unit will determine the number of loops necessary to machine the slot to size, again taking into consideration the need for each cutter pass to overlap to provide a completely clean surface. Another variation of this cycle will complete the looping sequence before dropping to the next Z depth.

Similar to the slot milling cycle is the pocket milling cycle. This cycle commences at the center of the pocket, the cutter feeding in the Z axis to a programmed depth. There follows a series of loops until the programmed X and Y dimensions are reached, again with a cutter overlap on each pass. If the pocket depth is such that more than one increment in the Z axis is necessary, the cutter is returned to the center of the pocket, Z increments down, and the cycle is repeated. Some systems provide for a cycle that roughs out the main pocket and then machines it to size with a small finishing cut. A pocket milling cycle is illustrated in Figure 8.73.

Figure 8.74 shows another widely used cycle referred to as a "bolt hole circle." This is for drilling a series of equally spaced holes on a pitch circle diameter. Given that the cutter has been brought to the pole position indicated, the other dimensional data required are the position of the first hole, the Z axis movement, the pitch diameter or radius depending on the control system, and the number of holes required. The control will make all the necessary calculations to convert the polar coordinates to linear coordinates and will effect slide movements accordingly.

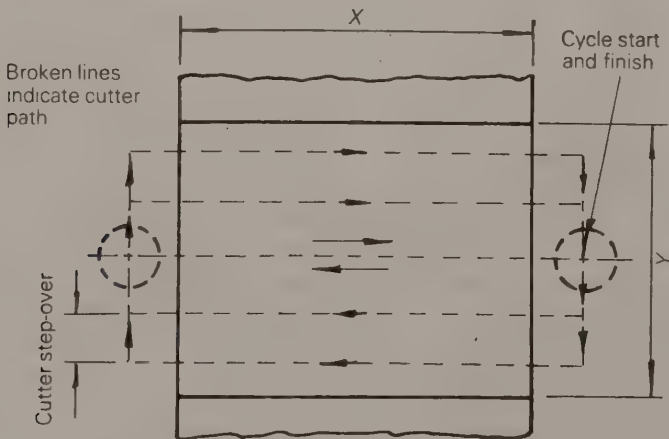


Figure 8.72 Slotting cycle.

Broken lines indicate
cutter path

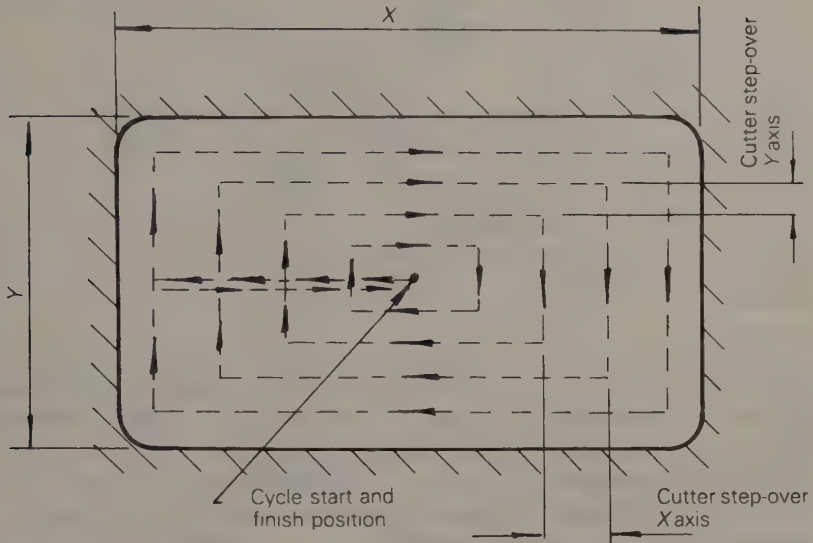


Figure 8.73 Pocket milling cycle.

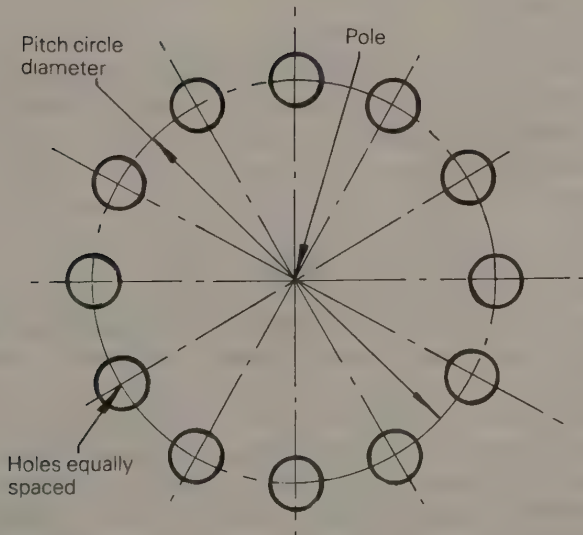


Figure 8.74 Bolt hole circle.

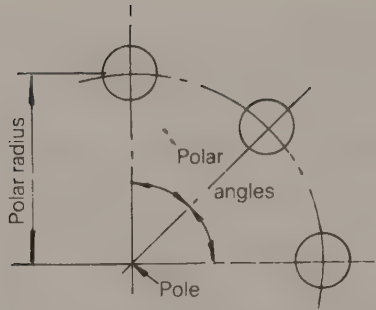


Figure 8.75 Positioning using polar coordinates.

A variation of this cycle will cater for just two or three holes positioned in an angular relationship to each other. An example is detailed in Figure 8.75. Again, the pole position is programmed and the cutter will be in this position when the cycle commences. The additional data that will be required are the Z axis movement, the polar radius and the polar angle(s); the controller will convert this information to slide movement in the relative axes.

Further milling cycles include those for boring, threading, elliptical profiles and even for the machining of helical arcs which simply requires data defining an arc in the X and Y axes and the change in the Z axis dimension between the start and end point. A moment's thought about the mathematical complexity of programming a cutter path such as this should be more than sufficient to emphasize how valuable canned cycles are as an aid to simplifying part programming procedures.

It should be noted that specialized cycles are normally meant for uniform or symmetrical shapes and as programming becomes more complex it is usually necessary to return to "G" code programming. Many conversational controls now have the ability to be programmed in two of three dimensional mode and conversational or "G" code mode.

ROTATION AND TRANSLATION

The positions of holes in angular relationship to each other was discussed previously on bolt circles. The machined feature—the hole—is rotated about a pole position.

The ability to rotate the positions of holes in this way is generally associated with the bolt hole circle facility and is commonly available. Many control systems also have the ability to rotate more complex features. The principle of rotation is illustrated in Figure 8.76.

A programming facility closely allied to rotation is translation. This permits the programmer to reposition a feature about a defined pole or center and then

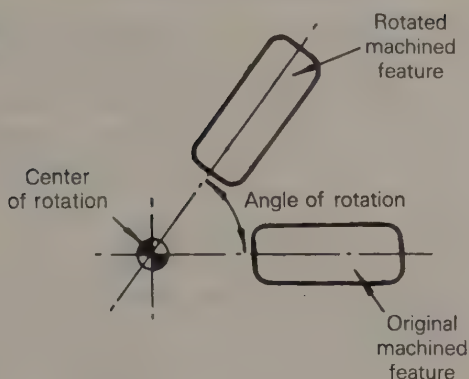


Figure 8.76 *Rotation of machined feature.*

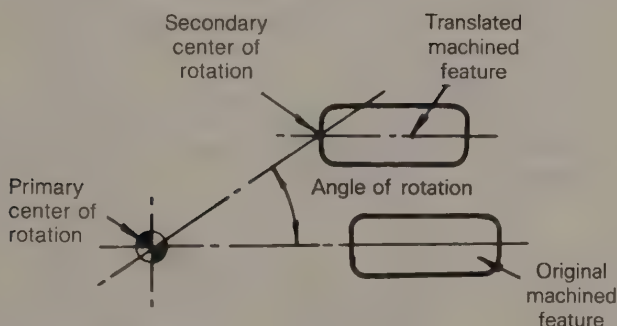


Figure 8.77 *Polar translation.*

to rotate the feature about a predetermined point on the feature itself. The principle is illustrated in Figure 8.77.

A bonus of the translation facility is that features of complex shape that are required in an angular location, which thus present some fairly complex programming calculations, can be programmed as though they lay on a true *XY* axis, simplifying the calculations required. They can then be repositioned using translation.

Translation may also be defined in linear terms and is, in effect, a datum shift facility. The shift may be in the *X* or *Y* axis or in both. Translation defined in this manner is illustrated in Figure 8.78.

SCALING

The scaling facility available on some control systems enables components that are geometrically identical but uniformly variable dimensionally to be produced from the same program data.

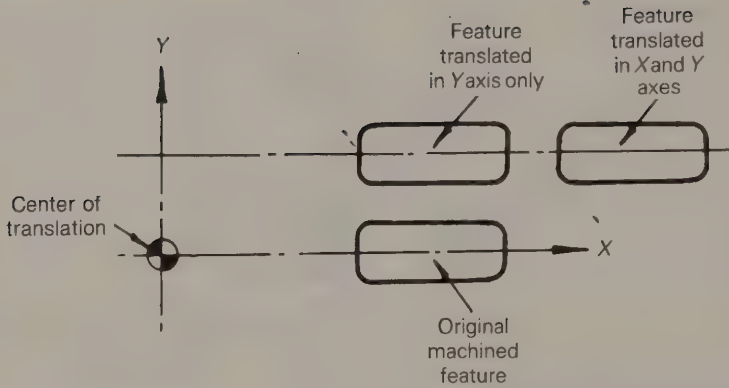


Figure 8.78 *Linear translation.*

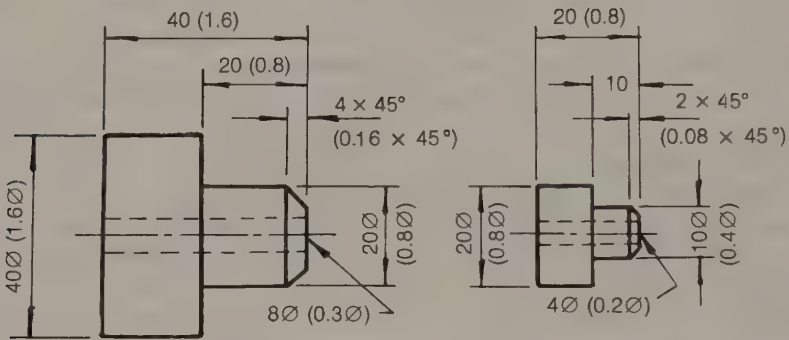


Figure 8.79 *Geometrically identical components suitable for production by scaling. (Inch units are given in parentheses.)*

Figure 8.79 illustrates two components, the production of which could be accommodated by scaling. Scaling is also available for milling operations. It can be applied to complete components or to one feature of a component.

An example of a scaling factor range, available on a widely used vertical machining center, is from 0.002 to 250. With such an extensive range the desired reduction or increase in size could well involve a machining requirement outside the capabilities of the machine, in which case an error message would be indicated. In practice components likely to be considered for production by scaling would rarely involve the use of widely varying scaling factors, but even a small scale factor, say 2, could produce an unacceptable result if the original data was for a fairly large component,

Some feel it is proper to use the lower end of a scaling factor range to minutely increase or decrease the machined size in one or more axes to maintain

a dimensional tolerance that may be being lost owing, for example, to the effect of clamping or distortion of the workpiece. This is not normally done, though, and is usually the task of the tool compensation option to be discussed subsequently.

MIRROR IMAGE

Mirror image is the term used to describe a programming facility used to machine components, or features of components, that are dimensionally identical but geometrically opposite either in two axes or one axis. By using the mirror image facility, such components can be machined from just one set of data.

In Figure 8.80 an original component feature is indicated in the bottom left-hand corner of the diagram. A complete mirror image of that feature is shown in the top right-hand corner, while mirror images in the X axis and the Y axis are shown, respectively, in the bottom right- and top left-hand corners of the diagram.

To produce a complete mirror image both the X axis and Y axis dimensional values will change from negative to positive. For half mirror images the dimensional values will change from negative to positive in one axis only.

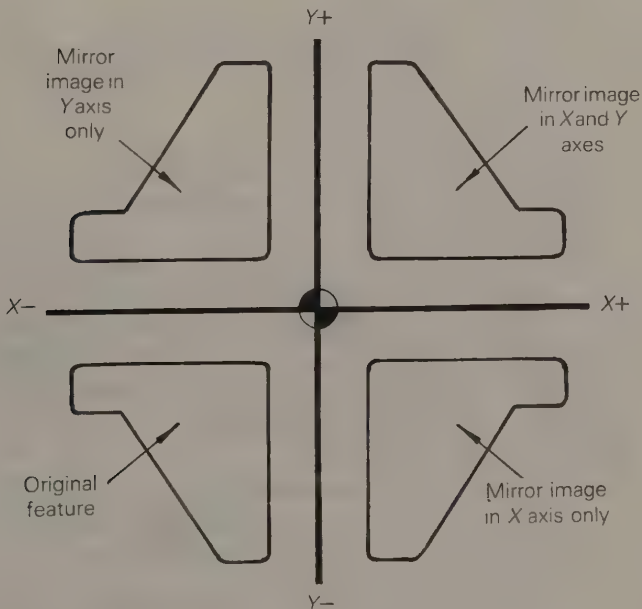


Figure 8.80 Mirror image.

CONVERSATIONAL PART PROGRAMMING EXAMPLES

The programming examples that follow were prepared to show the vast differences in systems as well as their similarities. Both examples are from vertical CNC milling machines. The positional calculations are not shown because they are computed in the same manner as in the manual part program examples given earlier in the chapter.

The first example (Figure 8.83) is that of a Wells Index 320 mill (Figure 8.81) with a Heidenhain TNC145 control (Figure 8.82). The example indicates some screen commands as well as operator responses and then a print out of the program. This example is intended to give a feel of what it is like to program this type of conversational control. (These examples are given in inch units only.)

Example 1

Operation description:

Using 0.500 in. diameter endmill cut a 0.100 in. deep step around the part maintaining a 0.250 in. width (Figures 8.83 and 8.84).

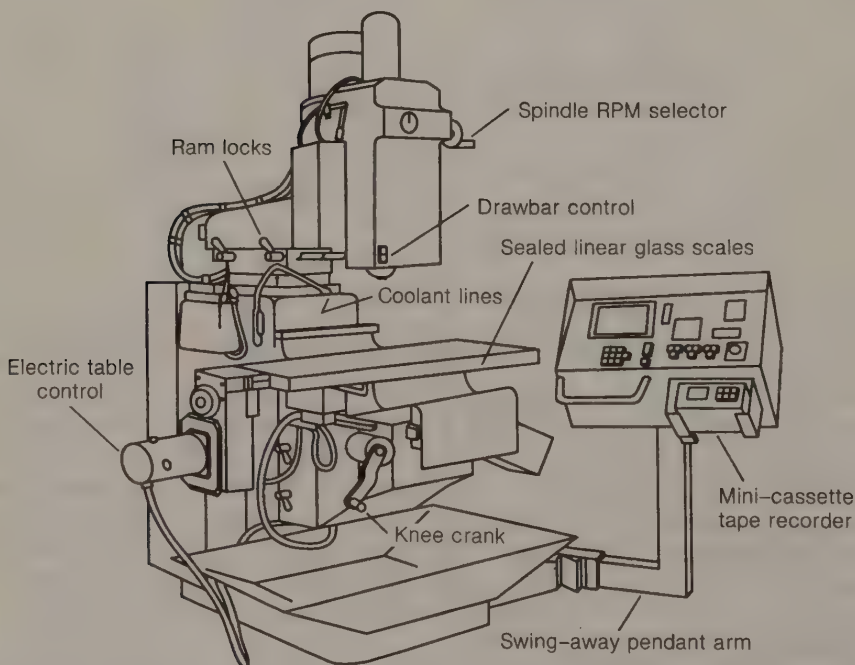


Figure 8.81 Wells-Index System 3 CNC mill with Heidenhain 145C control.

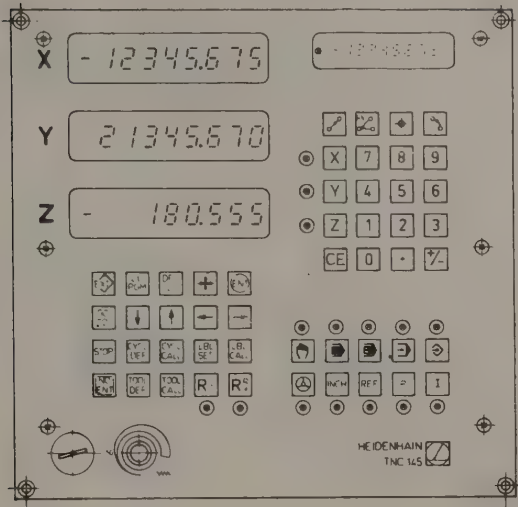


Figure 8.82 TNC 145C console keyboard.

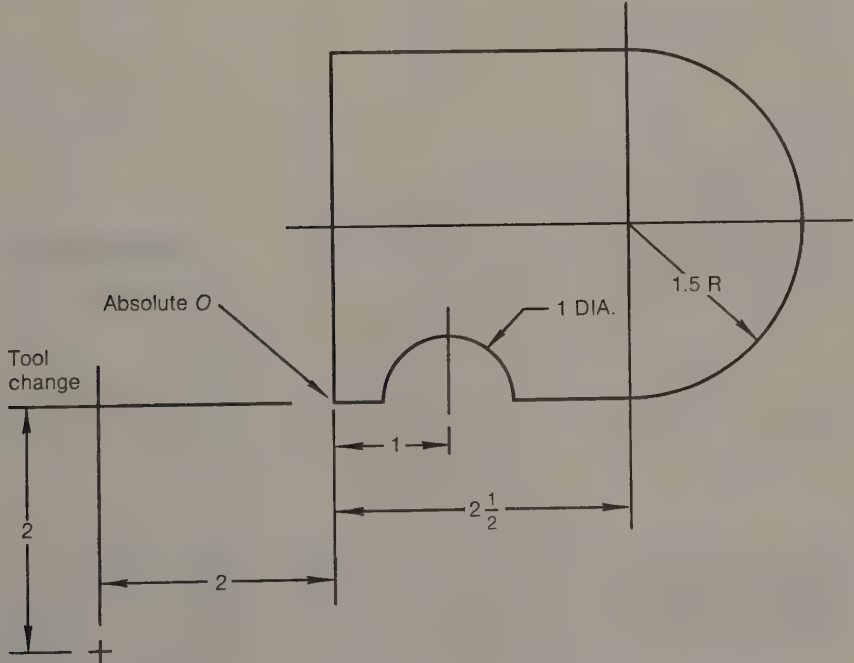


Figure 8.83 Wells example.

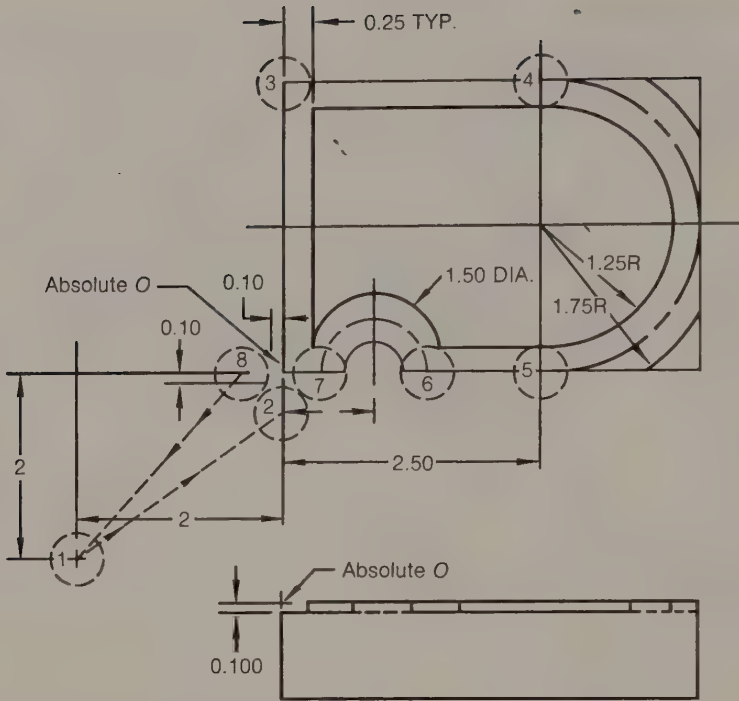


Figure 8.84 Wells cut exercise.

Wells Example of Program Inputs

Operation to be performed	Message on CRT	Operator keystrokes
With machine powered up, define tool to be used.	N1 TOOL NUMBER? TOOL LENGTH? TOOL RADIUS? BLOCK COMPLETE?	Tool Definition Button 1 Enter 0 Enter .25 Enter Enter
Call tool to be active for machining.	N2 TOOL NUMBER? WORKING SPINDLE AXIS X/Y/Z? SPINDLE SPEED S RPM? BLOCK COMPLETE?	Press Tool Call 1 Enter Z Enter 1600 Enter Enter
Move from tool change to position two on part.	N3 FIRST COORDINATE? SECOND COORDINATE?	Press Linear Motion X0 Enter Y-.350 Enter

Operation to be performed	Message on CRT	Operator keystrokes
	TOOL RADIUS COMP	
	RL/RR/NO COMP?	Enter
	FEED RATE?	220 Enter
	AUXILIARY FUNCTION	
	M?	03 Enter
	BLOCK COMPLETE?	Enter

The question answer process is continued until the program is completed.

Wells Example Program Read-out

N1 TOOL DEF 1	L + 0.0000		Define Tool #1
	R + 0.2500		
N2 TOOL CALL 1	Z S1600.000		Active Tool #1
N3 L X + 0.0000	Y-0.3500		Move to Pos. 2
		R0 F2200 M039	Spindle on.
N4 Z+0.1000			Z clearance
		R0 F1500 M08	Coolant On
N5 Z0			Z to depth
		R0 F250 M	
N6 Y+3.0000			
		R0 F250 M	Feed to Pos. 3
N7 X + 2.5000			Feed to Pos. 4
		R0 F250 M	
N8 CC X+2.5000	Y+1.5000		Define circle center
N9 C X+2.5000	Y0		Feed in circle to
		DR- F250 M	Pos. 5
N10 X+ 1.5000			Feed to Pos. 6
		R0 F250 M	
N11 CC X+1.0000	Y0		Define circle center
N12 C X+.5000	Y0		Feed in circle
		DR+ F250 M	to Pos. 7
N13 X-.350			Feed to Pos. 8
		R0 F250 M09	coolant off
N14 Z+2.0000			Retract Z
		R0 F1500 M05	Spindle off
N15 L X-2.0000	Y-2.0000		Move to tool
		R0 F2200 M02	change
			End of Program

Example 2 The second conversational control program example is for a Hurco CNC mill with the Ultimax control. This machine has dual CRT screens so that positional and program data can be reviewed on one and tool path graphics on the other.

The part example to be used for the Hurco program is the base component of a drill vise (Figure 8.85). The operations to be performed are the milling of a 1.375×3.837 pocket with a .500 endmill, the milling of a .500 slot with a .432 endmill, and the milling of eleven .125 radius adjusting notches with a .250 ball nose end mill. See tool graphics for tool change position and program zero location.

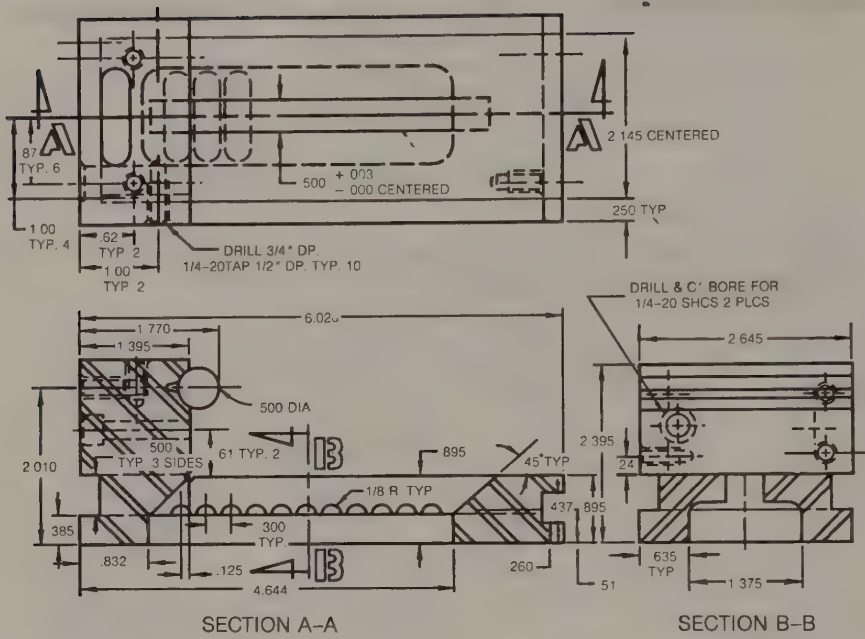


Figure 8.85 Base component drill vise.

Hurco Example of Program Inputs

Operation to be Performed

With machine powered up, define tool to be used (Figure 8.86).

Message on CRT

Upper screen displays machine status
Lower screen display reads tool setup display figure
Cursor at tool
Cursor at Type
Cursor at Diameter
Cursor at zero calibration

Cursor at speed

Cursor at coolant
Cursor at material
Cursor at flutes

Operator Key Strokes

Push tool setup soft key

Key in 1 and enter.
Push End Mill key
Key in 0.500
Jog tool down against reference point and push SET TOOL ZERO
Push TOOL UP key to retract spindle
Push CLOCKWISE key and then enter 0 speed to get control speed calculation
Push MIST key
Push HSS key
Enter number of flutes in cutter

		TOOL SETUP				
X	0.0000	0.0000	TOOL IN SPINDLE	0	DELETE TOOL	
Y	0.0000	0.0000	SPINDLE	0		
Z	0.0000	0.0000	FEED	0.0		
			MAGAZINE	0		
TOOL TYPE					> 1	
PITCH					0.000	
DIAMETER					0.0000	
ZERO CALIBRATION					0.0000	
SPEED (RPM)					CW 0	
COOLANT					OFF	PART SETUP
MATERIAL						
FLUTES					2	PART PROGRAMMING
						TOOL UP
						SET TOOL ZERO

Figure 8.86 Tool setup screen.

Operation to be Performed	Message on CRT	Operator Key Strokes
Repeat process for tools 2 and 3		
Position tool away from part (Figure 8.87).	Block 1 Cursor at TOOL	Push POSITION soft key Enter the tool number followed by the X and Y coordinates for the position
	Cursor at STOP	Enter NO by pushing button so machine does not stop at this position
	Cursor at INDEX PULSES	Enter 0 no index table is involved.
Create MILL FRAME (Figures 8.88 and 8.89)	Block 2	Press MILLING soft key to get main milling menu
	Choose type of block Cursor at TOOL Cursor at TYPE Cursor at FINISH PASS Cursor at X CORNER	Press FRAME soft key Enter 1—using key pad Press POCKET Press YES soft key Enter -.563—to establish lower left hand corner of pocket in X axis

BLOCK 2		MILL FRAME		1) ON
TOOL		1	Z START	0.0500
TYPE	>	ON	Z BOTTOM	-0.9500
FINISH PASS		YES	PLUNGE FEED	1.5
X CORNER	-0.5630		MILL FEED	8.0
Y CORNER	-0.6870		SPEED (RPM)	500
X LENGTH	3.8370		PECK DEPTH	0.1000
Y LENGTH	1.3750		(TOOL DIAMETER)	0.2500
CORNER RADIUS	0.2500			
				4) INSIDE TANGENT
				5) OUTSIDE TANGENT
				6) POCKET

Enter cutter placement

Figure 8.89 Mill frame.

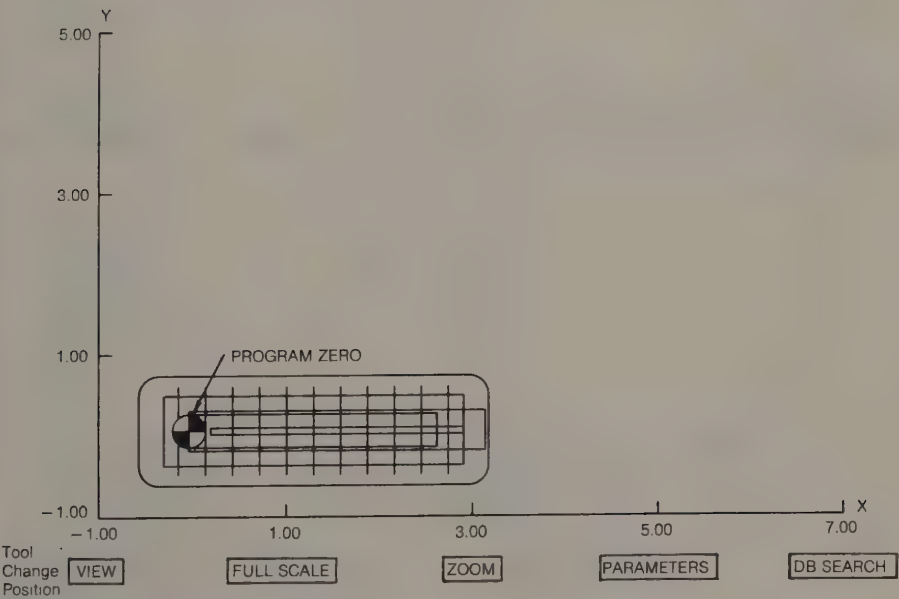


Figure 8.90 Mill pocket tool graphics.

Operation to be Performed	Message on CRT	Operator Key Strokes
	Cursor at Y CORNER	Enter -.687—to establish lower left hand corner of pocket in Y axis
	Cursor at X LENGTH	Enter 3.837—the "X" length of the pocket to be cut
	Cursor at Y LENGTH	Enter 1.375—the "Y" length of the pocket to be cut
	Cursor at CORNER RADIUS	Enter .250—the cutter radius
	Cursor at Z START	Enter .05—the starting depth of cut
	Cursor at Z BOTTOM	Enter -.950—the final depth of pocket
	Cursor at PLUNGE FEED	Enter 1.5—feed in IPM for plunging to depth
	Cursor at MILL FEED	Enter 8.0—feed in IPM for X and Y cutting moves
	Cursor at SPEED (RPM)	Enter 500—cutter rpm
	Cursor at PECK DEPTH	Enter .100—Z axis in feed increment for each pass.

This type of program entry process is continued until the entire program is complete.

Hurco Example Program Readout

HURCO ULTIMAX PART PROGRAM		INCH	STANDARD
PART SETUP			
PART ZERO X	7.0481	SAFETY WORK REGION	
Y	5.9507	Z TOP (+)	999.0000
		Z BOTTOM (-)	-999.0000
		X LEFT (-)	-999.0000
		X RIGHT (+)	999.0000
		Y FRONT (-)	-999.0000
		Y BACK (+)	999.0000
		MATERIAL	CARBON STEEL
PROGRAM PARAMETERS			
GENERAL	OVERRIDE LOCKOUT		OFF
	PROGRAM PROTECT		PARTIAL
	RETRACT CLEARANCE		99.0000 IN
	RAPID TRAVERSE		250.0 IPM
HOLES	BORE ORIENT RETRACT		0.0200 IN
	DRILL DWELL		0.5 SEC
	BORE DWELL		1.0 SEC

MILLING	BLEND OFFSET	0.1250 IN
	BLEND OVERLAP	0.1250 IN
	FINISH FEED	100%
	FINISH SPEED	100%
	FINISH XY	0.0050 IN
	FINISH Z	0.0050 IN
	MILLING DIRECTION	CONV
	POCKET OVERLAP	10%

TOOL SETUP

			(Tool Definitions)
1 TYPE	END MILL	COOLANT	MIST (Tool # 1
DIAMETER	0.5000	MATERIAL	HSS .500 End
ZERO CALIBRATION	1.2032	FLUTES	2 Mill)
SPEED (RPM) CW	0		
2 TYPE	END MILL	COOLANT	MIST (Tool #2 .432
DIAMETER	0.4320	MATERIAL	HSS end mill)
ZERO CALIBRATION	1.4859	FLUTES	2
SPEED (RPM) CW	0		
3 TYPE	END MILL	COOLANT	MIST (Tool #3
DIAMETER	0.2500	MATERIAL	HSS .250 ball
ZERO CALIBRATION	1.6097	FLUTES	2 nose end
SPEED (RPM) CW	0		mill)

DATA BLOCKS

1 POSITION			
TOOL	1	STOP	NO (Tool change
X	-2.0000	INDEX PULSES	0 .5 end
Y	-2.0000		mill)
2 MILL FRAME			
TOOL	1	Z START	0.0500 (Milling of
TYPE	POCKET	Z BOTTOM	-0.3850 1.375 x
FINISH PASS	YES	PLUNGE FEED	1.5 3.837
X CORNER	-0.5630	MILL FEED	8.0 pocket)
Y CORNER	-0.6870	SPEED (RPM)	500 (See
X LENGTH	3.8370	PECK DEPTH	0.1000 graphics
Y LENGTH	1.3750		of tool
CORNER RADIUS	0.2500		path,
			Figure
			8.90)
3 POSITION			
TOOL	1	STOP	NO (Tool #2
X	-2.0000	INDEX PULSES	0 change
Y	-2.0000		position)
4. MILL CONTOUR			(Mill .500
			slot)
SEGMENT 0	START		
TOOL	2	Z START	-0.3500
CUTTER COMP.	LEFT	Z BOTTOM	-0.9500
FINISH PASS	YES	PLUNGE FEED	1.5
X START	0.0000	SPEED (RPM)	500
Y START	0.0000	PECK DEPTH	0.0750
SEGMENT 1	LINE		
X END	0.0000	XY LENGTH CAL	0.2510

Y END	-0.2510	XY ANGLE CAL	* -90.000	
Z END	-0.9500	FEED	8.0	
SEGMENT 2	LINE			
X END	3.2490	XY LENGTH CAL	3.2490	
Y END	-0.2510	XY ANGLE CAL	0.000	
Z END	-0.9500	FEED	8.0	
SEGMENT 3	LINE			
X END	3.2490	XY LENGTH CAL	0.5020	
Y END	0.2510	XY ANGLE CAL	90.000	
Z END	-0.9500	FEED	8.0	
SEGMENT 4	LINE			
X END	0.0000	XY LENGTH CAL	0.3.2490	
Y END	0.2510	XY ANGLE CAL	180.000	
Z END	-0.9500	FEED	8.0	
SEGMENT 5	LINE			
X END	0.0000	XY LENGTH CAL	0.2510	
Y END	0.0000	XY ANGLE CAL	-90.000	
Z END	-0.9500	FEED	8.0	
5 POSITION				
TOOL	2	STOP	YES	(Tool #3
X	-2.0000	INDEX PULSES	0	change,
Y	-2.0000			.250 End
				Mill)
6 PATTERN LOOP LINEAR				
NUMBER	11	ANGLE	0.000	(Pattern loop
X DISTANCE	0.3000	DISTANCE	0.3000	definition
Y DISTANCE	0.0000			for .250
				notches)
7 MILL CONTOUR				
SEGMENT 0	START		-0.370	
TOOL	3	Z START	-0.5100	(Machining
CUTTER COMP.	NO	Z BOTTOM	1.5	moves for
FINISH PASS	YES	PLUNGE FEED		top notches)
X START	-0.1250	SPEED (RPM)	1200	
Y START	0.5370	PECK DEPTH	0.0300	
SEGMENT 1	LINE			
X END	-0.1250	XY LENGTH CAL	1.0740	(Machining
Y END	-0.5370	XY ANGLE CAL	-90.000	moves for
Z END	-0.5100	FEED	4.0	bottom
				notches)
8 PATTERN END				(End of
				pattern
				and
				program)

QUESTIONS*

- 1 Devise a simple diagram to illustrate the axes of movement of a vertical machining center and explain why it is necessary to redefine some of these movements as an aid to part programming.

- 2 Make a simple sketch to show the difference between absolute and incremental dimensional data.
- 3 How is the difference between an incremental and absolute value indicated on word address control systems that permit the use of either in the same part program?
- 4 What value should be programmed when a drawing states an upper and lower limit to a toleranced dimension?
- 5 Briefly explain the difference in data required by the three methods of circular interpolation which use I, J and K values.
- 6 Explain when the use of a programmer-devised sub-routine would be justified.
- 7 How does a 'macro' differ from other types of programmer-devised routines?
- 8 Describe the concept of 'parametric' programming and suggest when its use would be advantageous.
- 9 Describe the programming technique of 'point definition' and describe the type of situation where it could be used to advantage.
- 10 What is the advantage of using a datum shift within a program?
- 11 What is the purpose of the: Operation Schedule? Tool Sheet? Coordinate Sheet?
- 12 What are the five responsibilities of a part programmer related to tooling?
- 13 What is meant by the term word address programming?
- 14 What is meant by the term conversational programming?
- 15 Write a definition for the following terms: Point to Point; Linear Interpolation; Circular Interpolation.
- 16 What is the difference between datum shift and the program datum point.
- 17 Explain what a canned cycle is.
- 18 What is the major purpose of tool offsets?
- 19 Describe the difference between translation and rotation of program segments.
- 20 Describe two basic planning considerations which facilitate producing a part in the shortest possible time.
- 21 Explain what is meant by a 'floating zero' and state the advantages of such a facility.

- 22 With the aid of a simple sketch, or sketches, describe a situation where two programmed zero shifts would be required during the production of a turned part.
- 23 Assuming a situation where the part programmer is not in close contact with the machine shop, what documentation should be prepared to facilitate the transfer of essential information between the two activities of programming and production?
- 24 Explain how the tool offset facility can be used to program a series of cuts along a turned profile using the same programmed slide movements for each pass.
- 25 Suggest one method each for keeping (i) tool indexing time, and (ii) slide movement to a minimum and state why this should be a programming objective.
- 26 With the aid of a simple sketch describe what is meant by 'tool nose radius compensation' and describe a practical approach that can be employed to determine whether the required compensation is to the right or left of a profile.
- 27 Describe in detail the methods that can be used to prove a part program.

*For additional programming exercises that can be completed by students refer to Appendix C.

9

PART PROGRAMMING CALCULATIONS

Having read the previous chapters the reader should now be aware that the application of CNC technology to the production of machined parts requires considerable knowledge and ability.

In the first instance there is the practical expertise associated with process planning, work-holding, tooling, and so on. In this respect there is continuity between what was expected of the traditional machine shop craftsman and what is required of the machine shop technician involved with new technology. The advent of CNC has had a considerable effect on the way machining tasks are tackled, but the need for practical expertise remains much the same. Furthermore, this expertise can only be obtained by shop floor experience. It is assumed that students involved with CNC part programming will have acquired, or will be in the process of acquiring, this essential knowledge.

Second, the machine shop technician using modern technology is required to become familiar with the programming languages employed by the control units fitted to the machines he or she will be using. In addition, if computer-aided part programming facilities are to be used, then the technician must also become proficient in the application of these techniques. In the last few years those who have been involved in the education and training of others in the use of CNC technology have found that students, in general, rapidly master the use of programming systems. The most advanced aspect of the technology seems to be the one which is most quickly mastered and applied, but unfortunately its application is often marred by a lack of practical expertise and mathematical ability.

It is the mathematical ability of workshop technicians that is the third area of expertise that has to be considered. Mention was made earlier of the need for the part programmer to be able to carry out calculations associated with speeds and feeds, profile intersection points, arc centers, and so on. Competent manual part programming is not possible without a fairly well-developed mathematical ability and a sound understanding of geometric construction.

The mathematics involved in part programming are essentially practical in nature. It is assumed that readers of this text will have already developed these particular skills, and that they will be capable of carrying out the necessary calculations.

Thus it is not the purpose here to provide a text for the student who wishes to learn mathematical concepts, but rather to provide a means of revising certain areas that are of particular interest to the CNC part programmer, and to

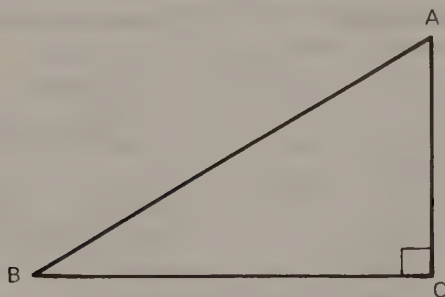
supply the student with examples that will develop the ability to deal with the mathematical elements of programming tasks. Reference to the geometry data contained below may be helpful in some cases.

While the following exercises are intended primarily for manual solution, a number of them will serve to introduce the facilities afforded by computer-aided part programming before the student attempts more comprehensive problems.

GEOMETRY DATA

The following information is provided for reference purposes.

Pythagoras' Theorem



$$AB^2 = AC^2 + BC^2$$

$$AB = \sqrt{AC^2 + BC^2}$$

Similarly,

$$AC = \sqrt{AB^2 - BC^2} \text{ and } BC = \sqrt{AB^2 - AC^2}$$

Example Figure 9.1 shows the detail of a milled component that has been dimensioned without regard to part programming needs. Dimension X is required. Using Pythagoras' theorem, calculate its value.

$$D^2 = 50.84^2 + 63.58^2$$

$$D = \sqrt{50.84^2 + 63.58^2}$$

$$= \sqrt{6627.2}$$

$$= 81.41$$

$$X = 81.41 + 10. = 91.41 \text{ mm}$$

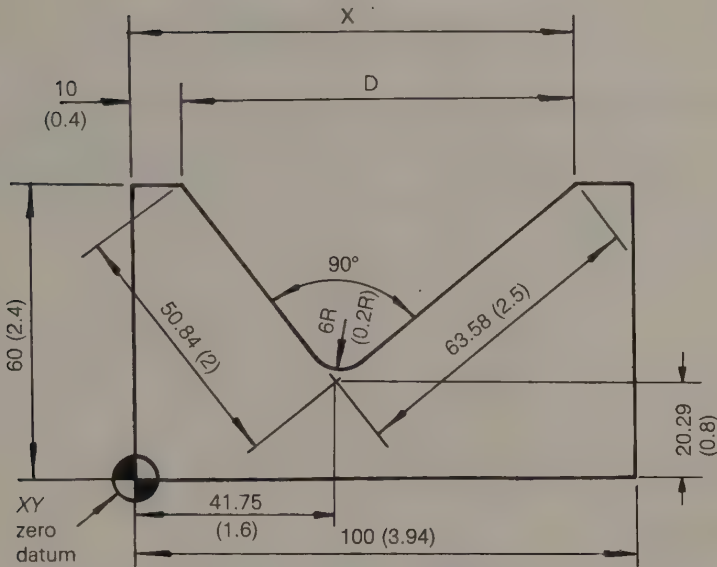
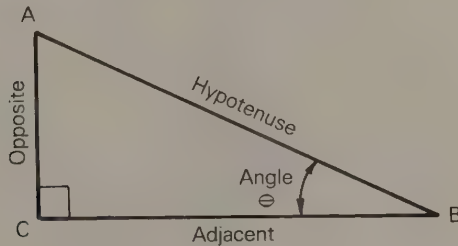


Figure 9.1 Using Pythagoras' theorem. (Inch units are given in parentheses.)

Trigonometrical Ratios



$$\text{Sine of the angle } \theta = \frac{\text{Opposite}}{\text{Hypotenuse}} = \frac{AC}{AB}$$

$$\text{Cosine of the angle } \theta = \frac{\text{Adjacent}}{\text{Hypotenuse}} = \frac{BC}{AB}$$

$$\text{Tangent of the angle } \theta = \frac{\text{Opposite}}{\text{Adjacent}} = \frac{AC}{BC}$$

Example Figure 9.2 shows the part detail of a component that is to be programmed with dimensional values being expressed in incremental mode. Thus the target position B has to be defined in relation to the end of the previous move A. Calculate the dimensions X and Y.

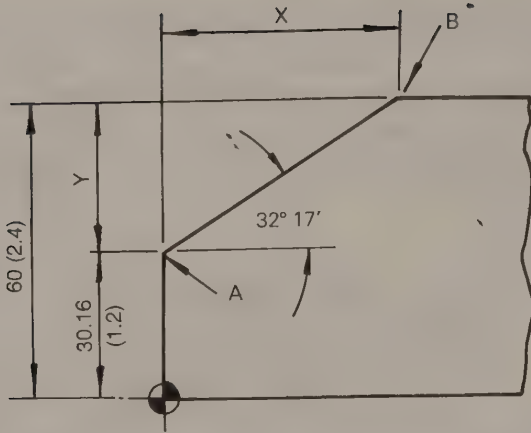


Figure 9.2 Using trigonometrical ratios. (Inch units are given in parentheses.)

To calculate Y

$$Y = 60. - 30.16 = 29.84 \text{ mm}$$

To calculate X:

$$\tan 32^{\circ}17' = \frac{\text{OPP}}{\text{ADJ}} = \frac{Y}{X}$$

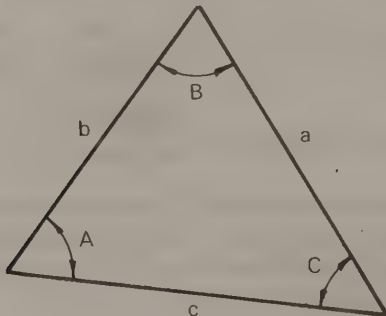
$$X = \frac{Y}{\tan 32^{\circ}17'}$$

$$= \frac{29.84}{0.6317}$$

$$= 47.23 \text{ mm}$$

The Sine Rule

For use with triangles that are not right-angled.



$$\frac{a}{\sin A} = \frac{b}{\sin B} = \frac{c}{\sin C}$$

Example Figure 9.3 shows a drawing detail of a machined feature. To facilitate part programming, calculate the dimensions X and Y.

To calculate the dimensions X and Y it is first necessary to determine ABD and the length AB from the information given. Note that $AB = c$.

$$\hat{ABD} = \hat{BAC} = 35^\circ$$

$$\hat{ACB} = 180 - (115 + 35) = 30^\circ$$

Using the sine rule to calculate AB:

$$\frac{b}{\sin B} = \frac{c}{\sin C}$$

$$\frac{b \times \sin C}{\sin B} = c$$

$$c = \frac{60. \times \sin 115^\circ}{\sin 30^\circ}$$

$$= \frac{60. \times 0.906}{0.5}$$

$$= 108.72$$

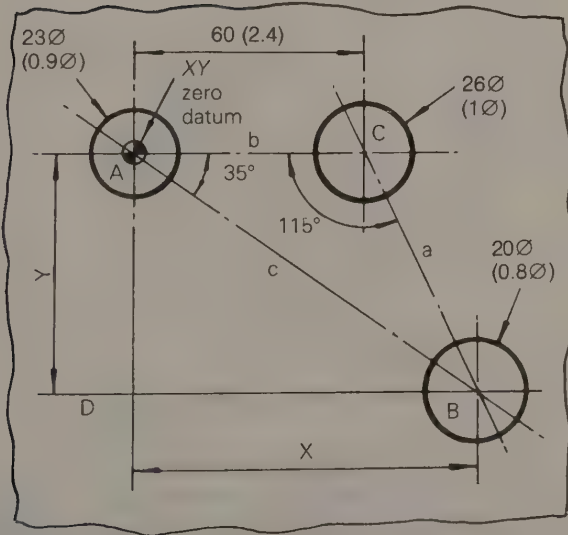


Figure 9.3 Using the sine rule. (Inch units are given in parentheses.)

To calculate dimensions X and Y using trigonometry:

$$\cos 35^\circ = \frac{\text{ADJ}}{\text{HYP}} = \frac{X}{AB} \quad \sin 35^\circ = \frac{\text{OPP}}{\text{ADJ}} = \frac{Y}{AB}$$

$$\cos 35^\circ \times AB = X \quad \sin 35^\circ \times AB = Y$$

$$X = 0.819 \times 108.72 \quad Y = 0.574 \times 108.72$$

$$= 89.042 \text{ mm} \quad Y = 62.405 \text{ mm}$$

EXERCISES

- Figure 9.4 shows details of a turned component dimensioned in a manner that does not accord with the part programmer's wish to locate the prepared billet against the backface of the chuck, this being the Z axis zero for the machine to be used, and to use absolute dimensional definition.

Make a half profile sketch of the component, and dimension it so that the process of preparing the part program will be simplified.

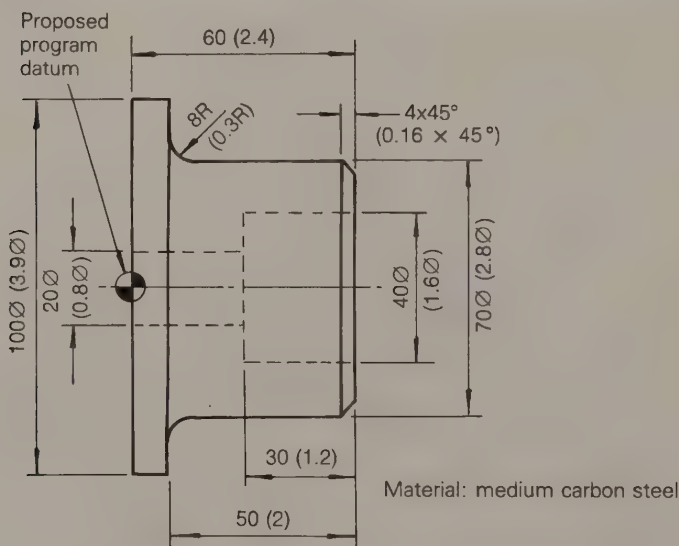


Figure 9.4 Exercise 1. (Inch units are given in parentheses.)

- Figure 9.5 shows details of a component that is to be milled and drilled on a vertical machining center. The programmer has decided to use the corner of the component as the program zero in both the X and Y axes. Make a sketch of the component and dimension it in a manner

that will be more convenient from a programming point of view, assuming that the programmer intends to use incremental positioning data.

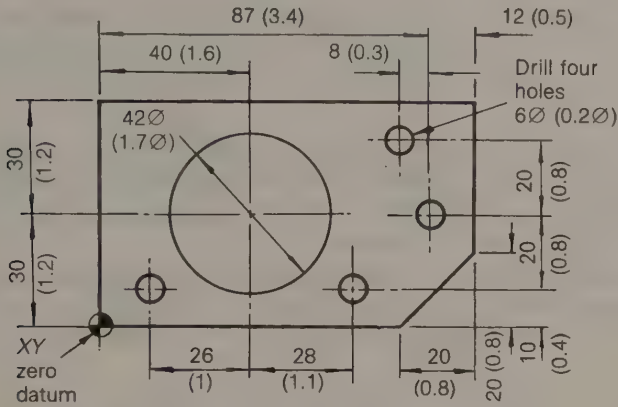


Figure 9.5 Exercise 2. (Inch units are given in parentheses.)

3. There is no obvious sequence in which to program the drilling of the six holes in the component shown in Figure 9.6. The programmer has initially chosen to adopt the sequence ABCDEF as indicated on the drawing. Since the time taken to actually drill the holes will be the same whatever sequence is used, the only time saving that can be made is by using the shortest possible positioning route.

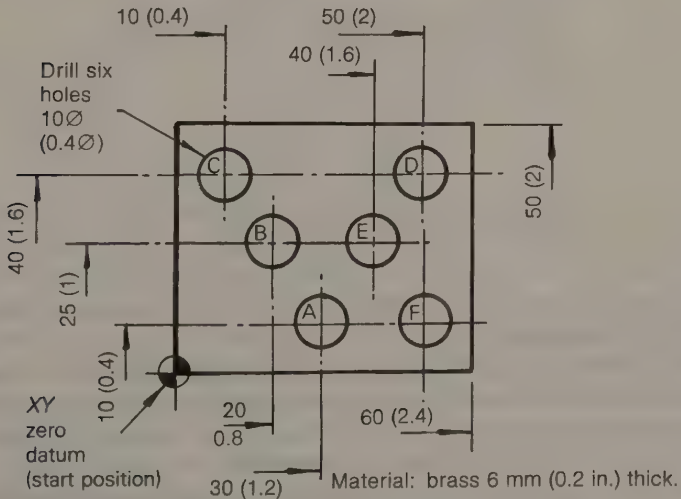


Figure 9.6 Exercise 3. (Inch units are given in parentheses.)

Using graph paper, draw the component accurately to scale, and check by measuring whether the proposed route is in fact the most efficient. If it proves not to be the shortest possible, state an alternative.

Given that the rapid feedrate for the machine is 2500 mm/min (100 in./min), estimate the total time saving that could be made by using this different positioning route, during a production run of 5000 components.

4. Figure 9.7 shows a component having a number of drilled holes. Assume that the starting point for the drilling operation is from a clearance plane 4 mm (0.2 in.) above the work surface and immediately above the XY zero datum indicated.

Accurately redraw the component on graph paper, and by scaling the drawing determine the most economical drilling sequence, taking account of the time taken in positioning.

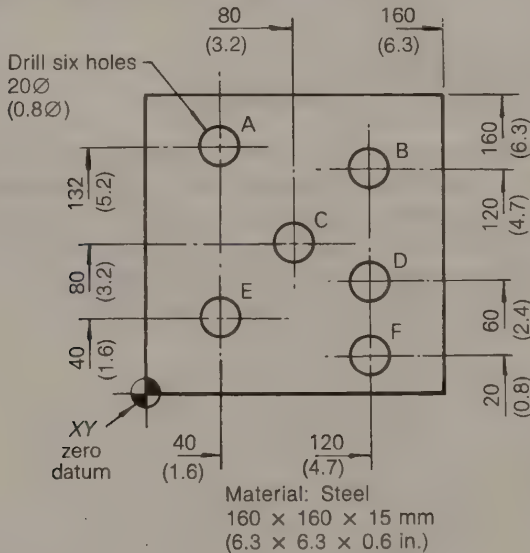


Figure 9.7 Exercise 4. (Inch units are given in parentheses.)

5. The milled impression shown in Figure 9.8 is to be machined on a vertical machining center.

Determine the shortest continuous tool path possible to machine the impression, assuming a start and finish position at zero in the X and Y axes. (Ignore movement in the Z axis as this is likely to be identical regardless of the route chosen in the X and Y axes.)

Calculate the time taken to complete the operation given that the rapid traverse rate for the machine is 4000 mm/min (158 in./min),

the feed rate for the metal cutting operation is 0.3 mm/rev (0.01 in./rev), the spindle speed is 3000 rev/min and the Z up position is 200 mm (8 in.) from the Z axis zero which is set at the top face of the work. Assume the tool change position is immediately above the XY axes zero. Normal Z axis clearance is 0.1 inch or 2 mm above the part face.

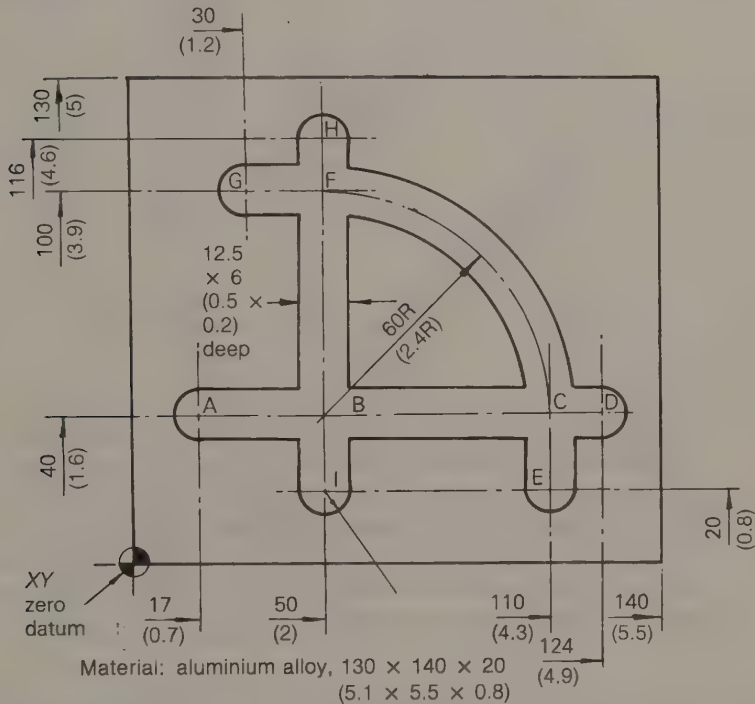


Figure 9.8 Exercise 5. (Inch units are given in parentheses.)

6. Assume that the feature shown in Figure 9.9, the machining of which involves a certain amount of stock removal prior to a finishing pass along the profile, is to be produced on a machine with a control system which does not possess a suitable canned cycle. Thus the programmer has no alternative but to determine, manually, the most efficient way of clearing the step. He or she may choose to program a series of cuts, with a small amount of cutter overlap to ensure a clean face, with the bulk of the stock being removed as the cutter travels along the X axis or, alternatively, along the Y axis.

Reproduce the pocket accurately to scale on graph paper and determine which, if any, of the two cutting directions would remove the

stock in the shortest time. Assume a cutter diameter of 12 mm (0.5 in.).

Express the movements required to remove the stock in terms of X and Y values, which could be incorporated as data in a part program.

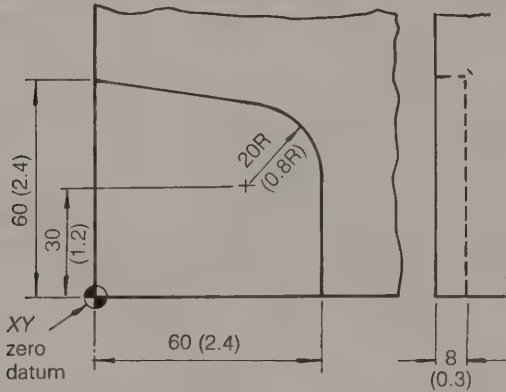


Figure 9.9 Exercise 6. (Inch units are given in parentheses.)

7. On a piece of graph paper, accurately reproduce the component shown in Figure 9.10 to a scale of twice full size and indicate on the drawing a series of roughing cuts, each having a depth of 4 mm (0.2 in.), that would reduce the bar sufficiently to leave approximately 1 mm (0.03 in.) along the profile for the final finishing cut.

Express the roughing cuts in terms of X and Z , that would enable them to be incorporated in the part program.

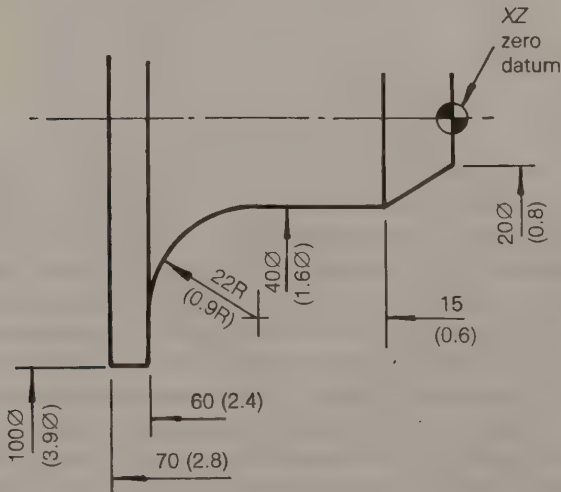


Figure 9.10 Exercise 7. (Inch units are given in parentheses.)

8. The angular feature of the component shown in Figure 9.11 is to be machined on a CNC-controlled machine, the control system of which does not include the cutter radius compensation facility. Calculate the incremental linear moves that will need to be included in the part program, assuming movement starts from and returns to the indicated datum. The diameter of the cutter to be used is 50 mm (2 in.), and there is to be approach and overrun distances of 4 mm (0.2 in.) as indicated.

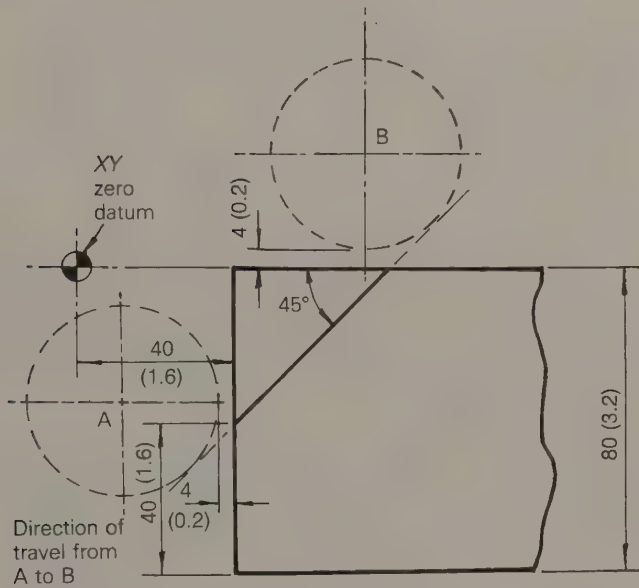


Figure 9.11 Exercise 8. (Inch units are given in parentheses.)

9. Calculate the incremental linear movements in the X and Z axes that will be required to finish machine the taper shown in Figure 9.12.

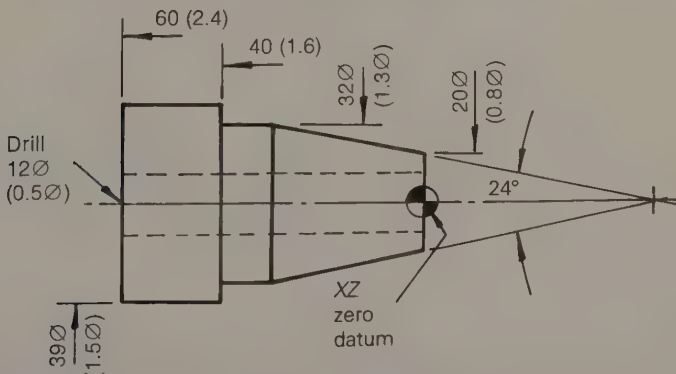


Figure 9.12 Exercise 9. (Inch units are given in parentheses.)

10. Figure 9.13 shows a feature of a turned component. Two of the dimensions necessary to complete a part program are not given. Determine which dimensions are missing and calculate their values.

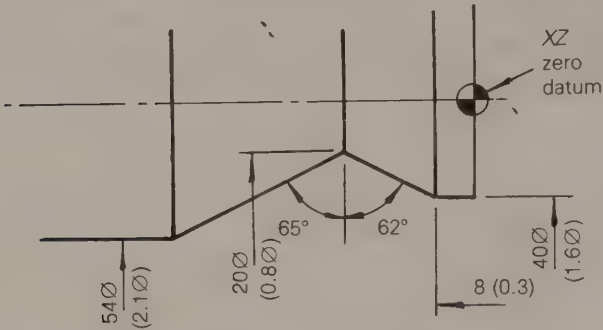


Figure 9.13 Exercise 10. (Inch units are given in parentheses.)

11. Figure 9.14 shows a feature of a component, namely, two parallel slots positioned at an angle of 52° to the X axis. The programmer intends to specify the numerical data for the required slide movements in absolute terms from the XY zero datum indicated on the drawing. To do this a number of calculations will have to be made and the drawing must be redimensioned.
- Carry out the necessary calculations and produce a second drawing of the component dimensioned in a manner that will be more appropriate.

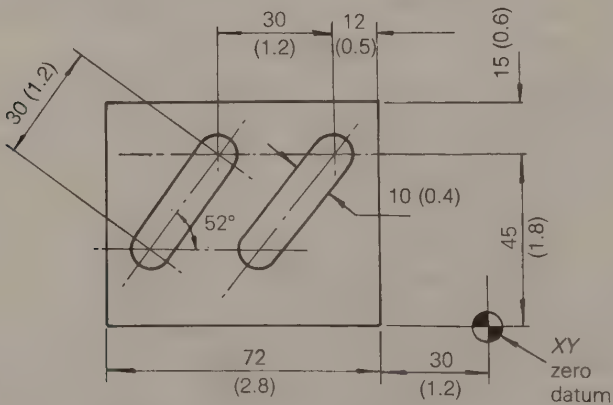


Figure 9.14 Exercise 11. (Inch units are given in parentheses.)

12. The component shown in Figure 9.15 is to have two holes drilled in the positions indicated. The programmer has opted to program slide

movements in absolute mode from the XY zero datum. In order to program in this way it will be necessary to calculate the linear coordinates for one of the holes.

Sketch the component, carry out the required calculations, and dimension your drawing accordingly.

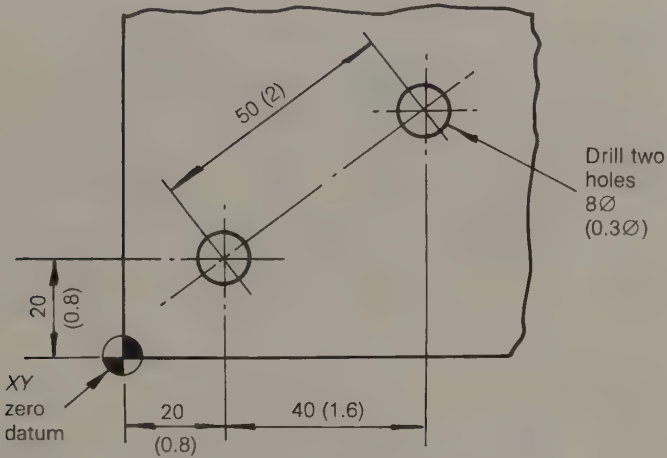


Figure 9.15 Exercise 12. (Inch units are given in parentheses.)

13. Calculate and list in programmable form the absolute coordinate dimensions in the XY axes necessary to drill the four holes shown in Figure 9.16 in the sequence ABCD, starting and returning to the program datum. Assume that a drill cycle controlling the depth to be drilled is already operative, and may be cancelled by programming G80.

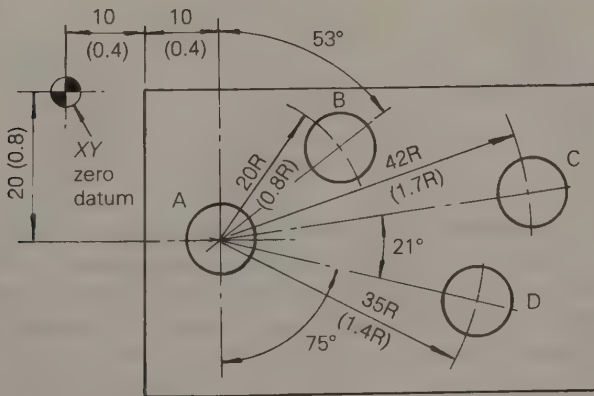


Figure 9.16 Exercise 13. (Inch units are given in parentheses.)

14. Figure 9.17(a) shows the position of three holes dimensioned in such a way that is not particularly helpful to the person preparing a part program, since the facility to program polar coordinates is not available. The preferred method of dimensioning is indicated in Figure 9.17(b). From the information provided, calculate the missing dimensions.

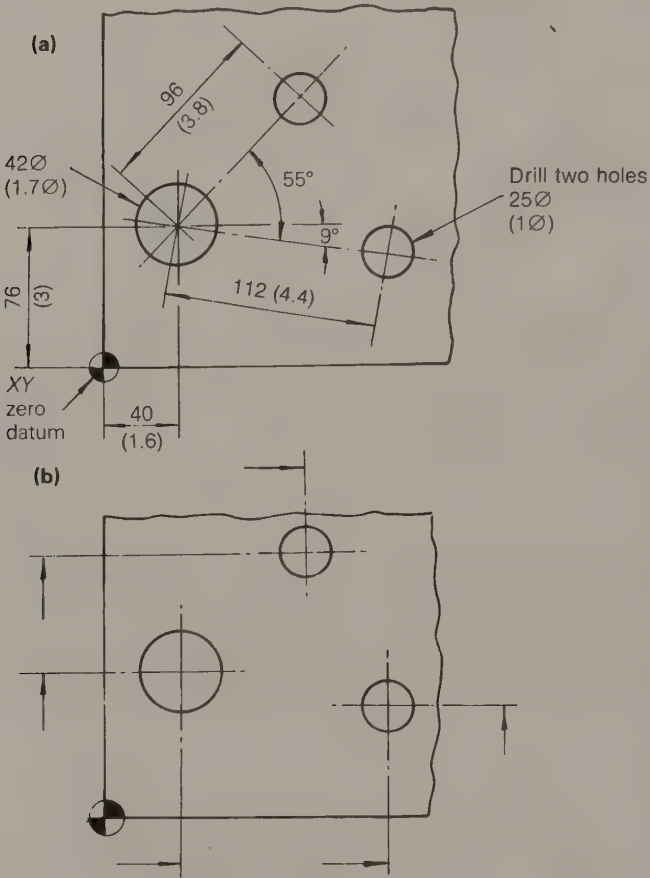


Figure 9.17 Exercise 14. (Inch units are given in parentheses.)

15. The four holes shown in Figure 9.18 are to be drilled in the sequence ABCD with the necessary slide movements expressed in absolute terms. Assuming a Z up travel of 200 mm (8 in.) immediately above the program datum to accommodate a manual tool change, and that a G81 drill cycle (cancelled by a programmed G80) is available, carry out the following:
- From the data stated on the drawing determine the dimensional value of the required slide movements.

List the data, in word address format, that would be required to complete the machining, assuming that all speeds, feeds etc. are already operative.

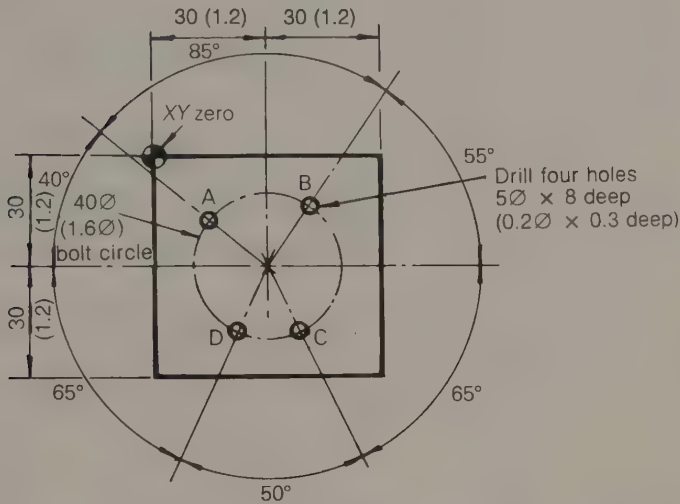


Figure 9.18 Exercise 15. (Inch units are given in parentheses.)

16. Eight holes are to be drilled on a bolt circle diameter as shown in Figure 9.19. The holes are to be drilled in the sequence A to H immediately after the central hole has been drilled. Taking the position of the central hole as the zero datum in both the *X* and *Y* axes carry out the following:

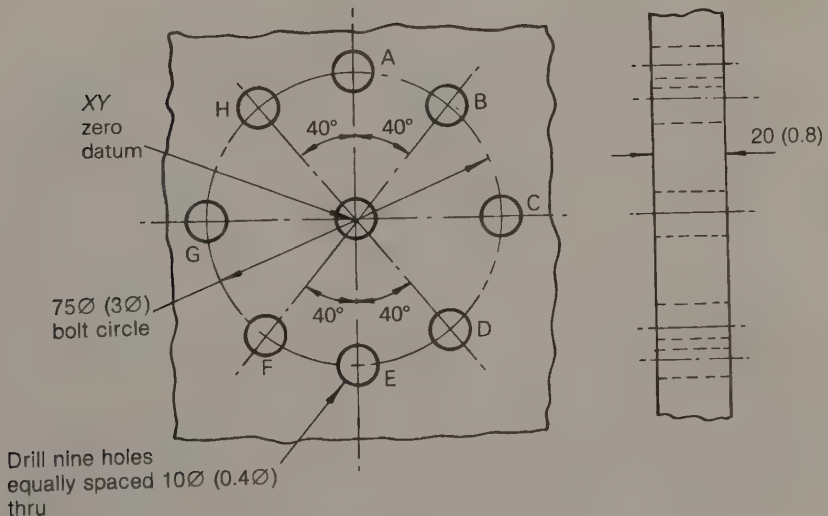


Figure 9.19 Exercise 16. (Inch units are given in parentheses.)

Calculate the dimensional value of the required slide movements, to facilitate programming in incremental mode.

List the data as they would be presented in a word address program, assuming a Z datum clearance of 2 mm (0.1 in.) and an excess travel of 5 mm (0.2 in.) on breakthrough. Assume that a spindle speed and feed rate have already been programmed and that a G81 drilling cycle is to be used.

17. The positions of three holes are given on a drawing as shown in Figure 9.20. The holes are to be drilled in the sequence ABC and the previous program data have brought the machine spindle into vertical alignment with hole A. Calculate the linear values of the incremental moves to be included in the part program to control slide movement in the X and Y axes.

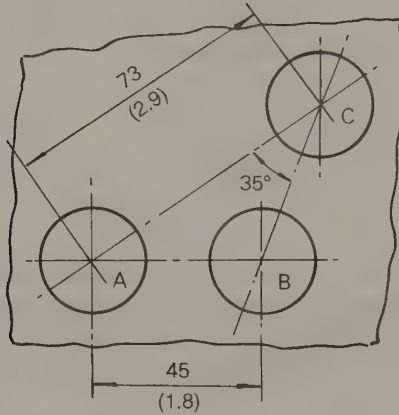


Figure 9.20 Exercise 17. (Inch units are given in parentheses.)

18. Complete the program data given below (for metric values only) for achieving relative cutter movement from P1 to P2 on the profile illustrated in Figure 9.21. The circular arcs are to be programmed by defining the target positions using X and Y values, and the arc centers are defined in relation to the starting point of the arc using I and J values.

```

N60 G01 Y-25
N70 G03
N80 G01 X20
N90
N100 G02
N110 G01 X10
N120
N130 G03
N140 G01

```

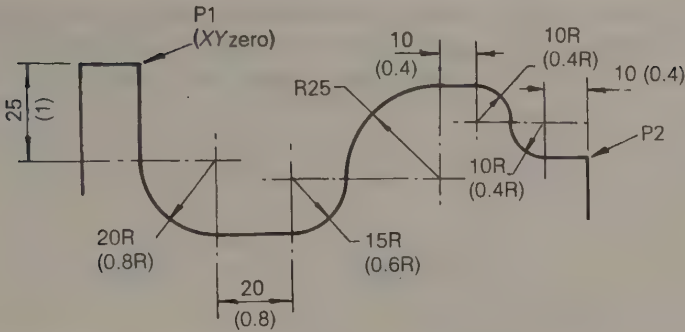



Figure 9.21 Exercise 18. (Inch units are given in parentheses.)

19. The turned component shown in Figure 9.22 is to be produced on a turning center where arc centers are defined in relation to the program datum using the address letters I and K. Complete the program block below in incremental terms (in metric units only):

N 025 G02 X-- Z-- I-- K--

Repeat the above exercise with the data expressed in absolute terms.

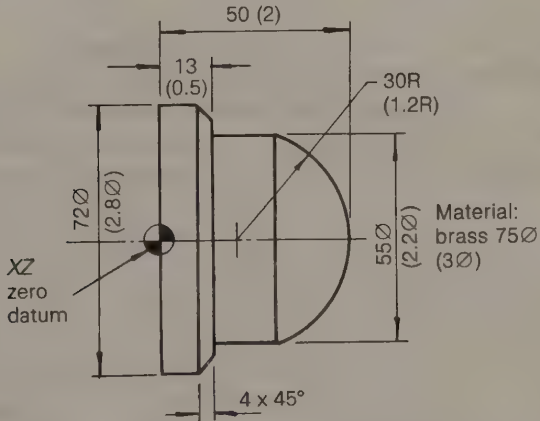


Figure 9.22 Exercise 19. (Inch units are given in parentheses.)

20. The concave arc of the turned component shown in Figure 9.23 is to be produced on a lathe where the control system will simply require a data input stating the radius of the arc, the direction of rotation, and the target position defined by X and Z values. Assume that the 8 mm (0.3 in.) diameter hole has already been drilled, and that the curve is to be produced by making a cutting pass starting from the center of the component and working outward. Complete the data necessary to

machine the feature, with the tool starting and finishing in the positions indicated. Assume that a suitable cutting speed and feed rate, and also cutter radius compensation, are already active.

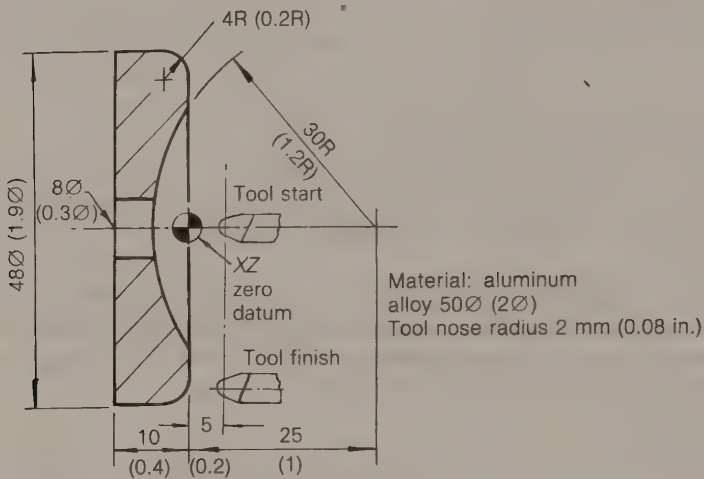


Figure 9.23 Exercise 20. (Inch units are given in parentheses.)

21. A cutout of 30 mm (1.2 in.) radius is to be machined in a 40 mm (1.6 in.) diameter disk as shown in Figure 9.24. The cutter start and finish positions giving a suitable approach and runout are indicated. Assume that cutter radius compensation mode will be operative and that the circular interpolation will involve two separate data blocks, one for

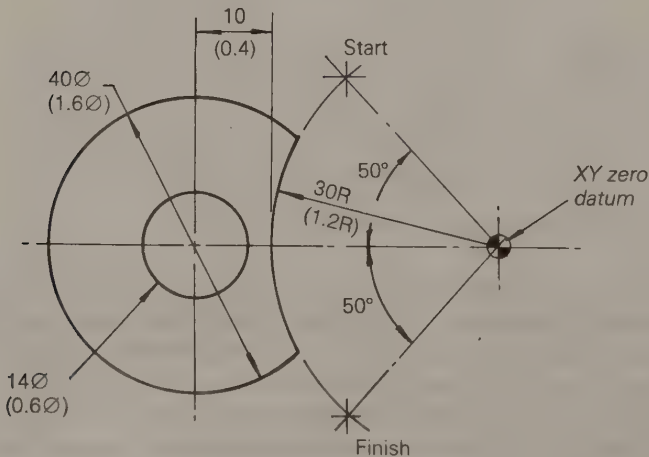


Figure 9.24 Exercise 21. (Inch units are given in parentheses.)

each quadrant. Determine the two data blocks that will be required, defining the arc start point in relation to the center using I and J values. Express the positional data in absolute terms.

22. Figure 9.25 shows a radial slot that is to be machined using a relative cutter movement from the start point to the finish point as indicated.

Calculate the target position in relation to the indicated program datum.

If the start position is to be defined in relation to the arc center using I and J definition, state their numerical values.

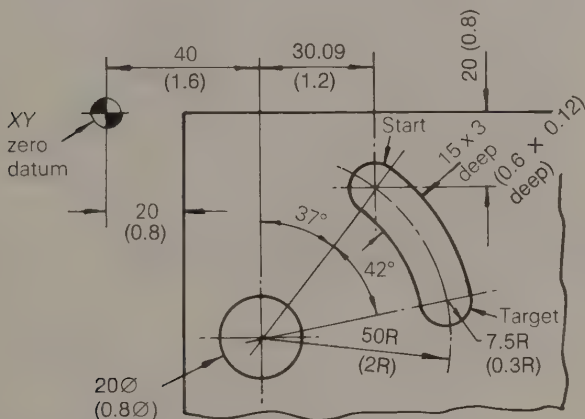


Figure 9.25 Exercise 22. (Inch units are given in parentheses.)

23. The two radial slots shown in Figure 9.26 are to be produced on a vertical machining center, with the starting points as indicated and the relative cutter movement being in a clockwise direction. The control system requires arc centers to be defined in relation to the arc starting point.

Determine the additional dimensional data, in absolute terms, that will be necessary for programming purposes in order to machine the slots.

24. The turned component in Figure 9.27 is to be produced on a machine fitted with a control system that requires the circular interpolation data entries to be programmed as follows:

- The target position using X and Z numerical values.
- The arc center in relation to the program datum using I and K numerical values.

Assuming that the program is to be compiled using absolute dimensions, complete the following N125 G02 X-- Z-- I-- K-- program entry for machining the arc.

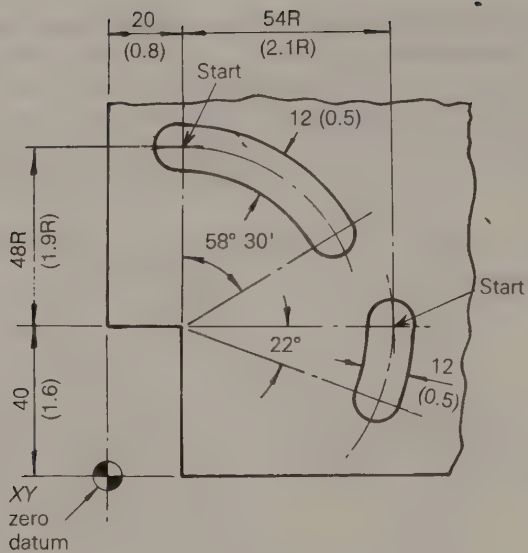


Figure 9.26 Exercise 23. (Inch units are given in parentheses.)

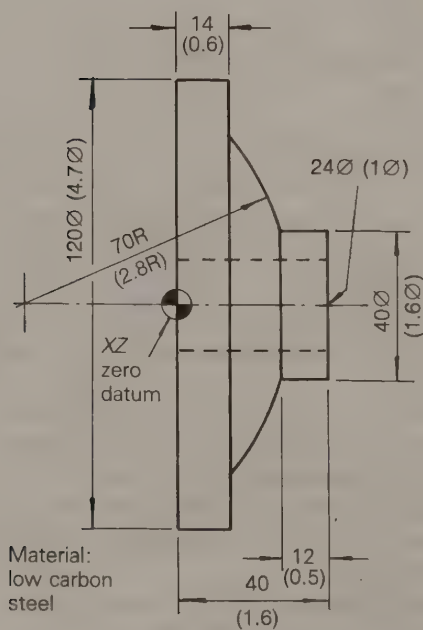


Figure 9.27 Exercise 24. (Inch units are given in parentheses.)

25. Figure 9.28(a) gives the details of part of a turned component. Figure 9.28(b) shows the same feature but with an indication of the dimensioning method that would be more appropriate since the machine to be used does not cater to multiquadrant programming.

Calculate the alternative dimensions required and make a redimensioned sketch of the component feature.

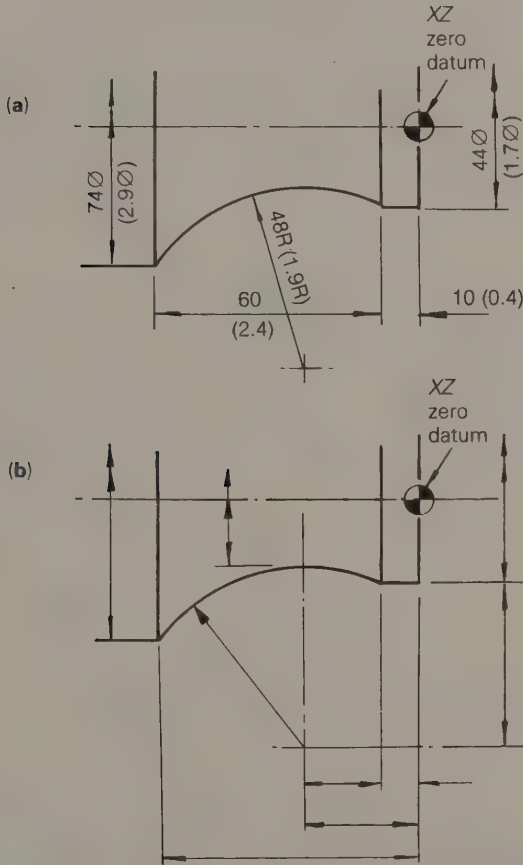


Figure 9.28 Exercise 25. (Inch units are given in parentheses.)

26. Figure 9.29 shows part of the profile of a milled component. The profile consists of four blending arcs of varying dimensions. In order to program the necessary circular moves the programmer will need to know the exact points at which the curves intersect with each other.

Construct accurately on graph paper the given profile and locate on your drawing the points of intersection.

Calculate and define numerically each intersection point.

Given the following programming information, list the program data necessary to achieve a cutter path that would machine the profile, commencing with linear movement from the XY zero datum and ending at point B.

Programming information: `

- G01 Linear interpolation, programmed feed rate.
- G02 Circular interpolation, clockwise.
- G03 Circular interpolation, counterclockwise.

Define all target positions incrementally and the arc centers in relation to the arc starting points using I and J values. Assume spindle control data, feed rates, etc., have already been programmed and that cutter radius compensation is active.

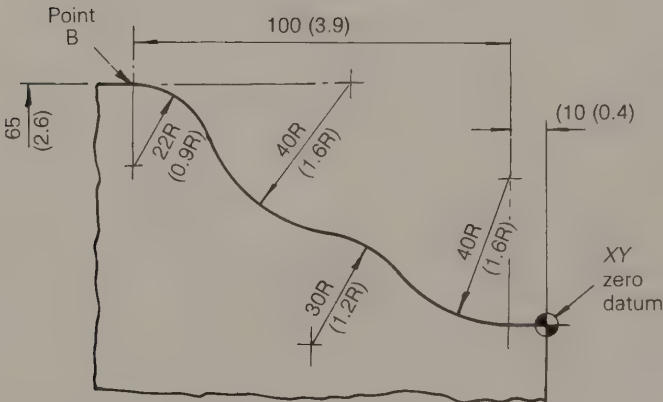


Figure 9.29 Exercise 26. (Inch units are given in parentheses.)

27. The milled profile shown in Figure 9.30 is to be produced on a machine with a control system that requires numerical definition of the target positions of all slide movements using the address letters X and Y, and arc centers to be defined in relation to the start of the arc using address letters I and J. Also, the system is not capable of multiquadrant programming, so movement in each quadrant must be programmed separately even when the same radius passes from one quadrant into a second.

Accurately construct the profile and indicate in absolute terms in relation to the program zero all intersection points and also arc start positions in relation to the arc centers.

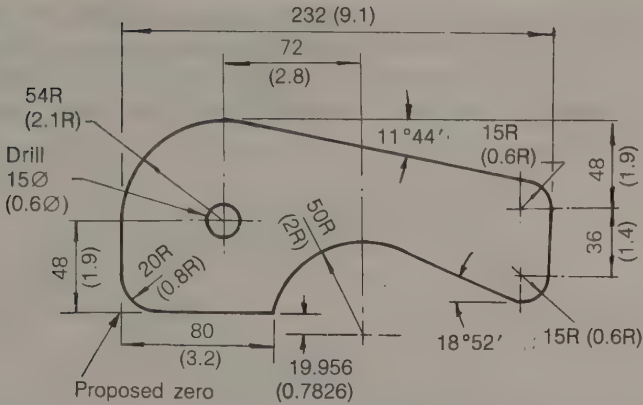


Figure 9.30 Exercise 27. (Inch units are given in parentheses.)

28. The internal profile of the component shown in Figure 9.31 is to be produced on a vertical machining center. Calculate and indicate on an appropriate sketch the additional dimensions that will be required to define the profile intersection points.

State which common programming facility would be suited to facilitate machining the profile.

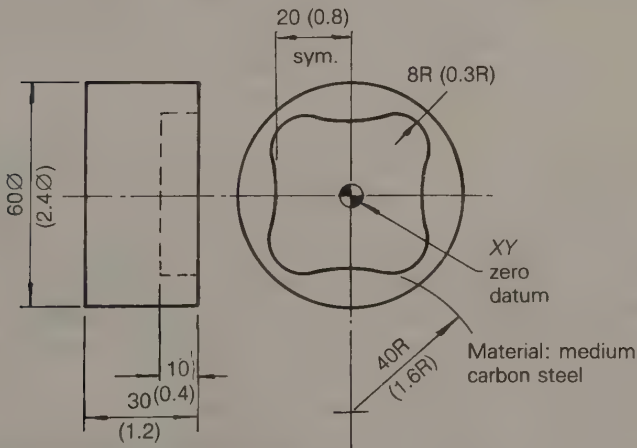


Figure 9.31 Exercise 28. (Inch units are given in parentheses.)

29. Figure 9.32 shows details of a milled component.

Draw the raised profile accurately to size and indicate on your drawing the arc centers and the intersecting points of the profile.

Calculate all the profile intersecting points required to complete a part program and dimension your drawing accordingly. Assume that arcs can only be programmed for one quadrant in any one block, with the arc centers being specified in relation to the program zero using I and J values.

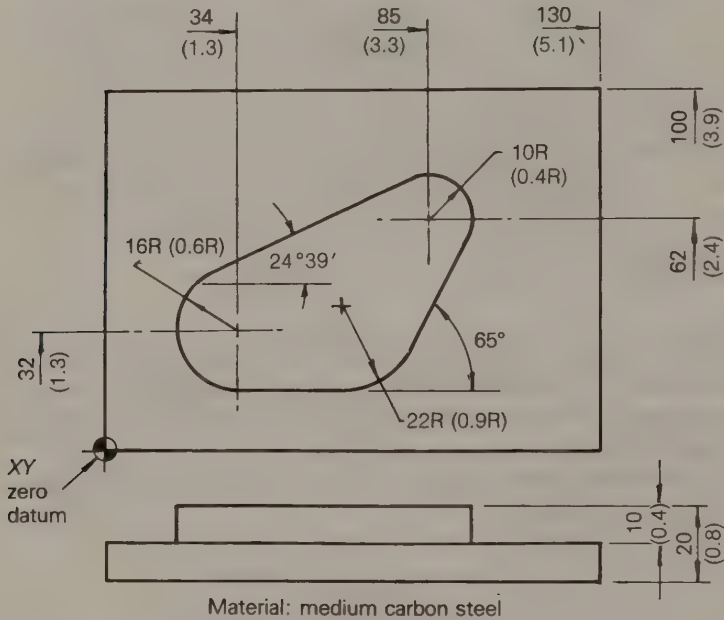


Figure 9.32 Exercise 29. (Inch units are given in parentheses.)

Refer to the cutting speeds and feeds given in the Appendices for appropriate data to complete the following questions.

30. Calculate a suitable spindle speed for drilling 6 mm (0.2 in.) diameter holes in a medium carbon steel using a high speed steel drill.
31. Determine a suitable spindle speed for face cutting T6 aluminum using a shell end mill of 40 mm (1.6 in.) diameter with cemented carbide insert teeth.
32. What increase in spindle speed would be appropriate if, when drilling a low carbon steel, a 20 mm (0.8 in.) diameter high speed steel drill is replaced with another having cemented carbide brazed tips?
33. Select a suitable cutting speed for turning a complex profile from brass using a cemented carbide insert turning tool. Given that the component is of variable diameter along its length, why would it be preferable to program a constant surface cutting speed rather than a set spindle speed?

34. Calculate the spindle speed to be used when milling grey cast iron using a 20 mm (0.8 in.) diameter cemented carbide insert end mill. If the speed calculated eventually proved to be too high, what action should be taken by the machine operator to rectify the situation?
35. Calculate suitable spindle speeds for roughing and finishing cuts when turning T6 aluminum, using cemented carbide tooling on a component having a nominal diameter of 38 mm (1.5 in.).
36. Select a suitable feedrate in mm/rev for turning medium carbon steel, using cemented carbide insert tooling.
37. Calculate a suitable feedrate in mm/min or in./min for a light turning operation on a stainless steel component having a diameter of 75 mm (3 in.), and when using cemented carbide tooling.
38. Given that an appropriate feedrate per tooth for face milling a low carbon steel is 0.3 mm (0.01 in.) per tooth when using cemented carbide tooling, what would be a suitable program entry in mm/min or in./min when using a 75 mm (3 in.) diameter cutter having six teeth.
39. The profile shown in Figure 9.33 is of part of a component that is to be machined from brass. The feed rate is to be programmed in mm/rev (in./rev) and the spindle speed controlled by programming a constant surface cutting speed in m/min (ft/min).

Select appropriate values for cutting speed and feed to be included in the part program, when using cemented carbide insert tooling.

Using the above combination of data, would the resulting surface finish be identical for both the parallel surfaces and the taper? If not, and assuming that a variation is not acceptable, how could the program be modified?

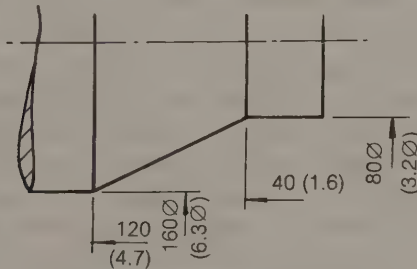


Figure 9.33 Exercise 39. (Inch units are given in parentheses.)

40. A face milling operation on aluminum alloy is to be carried out using a nine-cartridge cutter of 100 mm (4 in.) diameter and having cemented carbide inserts. Determine a suitable spindle speed and feed rate in mm/rev (in./rev) for inclusion in the part program.

10

COMPUTER-AIDED PART PROGRAMMING

THE APPLICATION AND ADVANTAGES OF COMPUTER-AIDED PART PROGRAMMING

When NC, as opposed to CNC, was first introduced, the only way a curve could be produced was by approximation, making a series of angular moves using slide movements in two axes that, when blended together, approximated to the curve required. The larger the number of small angular movements made, the more precise the final curve.

On the face of it this would appear to be a very reasonable solution to the problem of producing a curve until the work involved in making the necessary calculations is considered, not to mention the mathematical ability required. It was the sort of situation where a little computer help with the calculations was much appreciated. The more complex the profile—imagine an elliptical path, for instance—the more essential computer help became.

Today, thanks to the inclusion of a microcomputer as an integral part of even the most basic CNC control systems, the programming of a constant radius curve is a very simple matter indeed, often requiring nothing more than dimensional definition of the target position and the value of the radius. Even the more complex elliptical profiles can be programmed on some control systems simply by defining the major and minor axes.

The reader will recall that the programming of radial cutter paths is referred to as “circular interpolation,” while the facility used to program an elliptical profile is known as a special canned or program cycle.

There is a wide variety of canned cycles currently available with modern machine controllers. A number of these were described in Chapter 8. All of these canned cycles were designed and included in the control system with one objective in mind, that is, to simplify programming.

Canned cycles cater to sequences that are likely to recur regularly. But not all complex profiles regularly recur and when such a profile does occur, it can present problems as difficult, and possibly more so, as the problems associated with curves in the early days of NC.

In particular, the machining of complex profiles, or rather the preparation of the part program to achieve the machining, means that fairly complex calculations have to be performed to determine the geometry intersection points.

There is also the problem of determining efficient cutter paths to remove stock, and that can also be a laborious business.

It is to meet these requirements that special computer-based programming systems have been developed. The process of using these systems is referred to as Computer-Aided Part Programming, generally referred to as CAPP. (Note: the initials CAPP are also used in production engineering to denote Computer-Aided Process Planning, which is concerned with the total organization of a production operation of which Computer-Aided Part Programming may be an included activity.)

CAPP provides for a simpler, quicker, and more accurate approach to preparing a CNC part program. But at the same time it represents a considerable capital outlay that has to be justified by an eventual increase in efficiency and productivity.

The CAPP process involves preparing a program using a specially developed language, entering the resulting data into a computer and receiving back from the computer a program presented in a format acceptable to the machine controller. During the process the computer will have processed the data to verify their validity and, where necessary, performed computing tasks (many that would have been mathematically difficult and/or time-consuming if attempted manually), and then processed the results into machine language.

CAPP can, in the hands of experienced users, provide rapid programming solutions for the most difficult of work. Even for very simple work, where manually prepared programs could be produced fairly quickly, the use of CAPP is still a viable proposition.

A trend among engineering contractors is to ask the potential customer for a drawing of the component that is to be produced and then to program, using CAPP, and then manufacture a component. The result is then returned to the customer with the quotation, indicating very effectively the speed and quality of the service they offer. At the same time the contractor is able to cost the contract precisely, since it will be known exactly how long it took to machine the sample. This process is normally referred to as a customer bench mark.

A further advantage of CAPP is that the program is prepared and initially proven "off-line," that is, away from the machine. Data transfer electronically to the machine is available and rapid so there is no delay in getting a machine back into production following job changes.

COMPUTER INSTALLATIONS

CAPP systems are available for use on all types of computer installations from mainframe to micro.

The large mainframe computers possess the greatest computing power and their use is indispensable for very complex programming requirements. Unfortunately they are extremely costly and their installation is only economically

viable in large organizations where the computing needs—not only for CAPP but for a whole range of industrial and commercial activities—are considerable.

A mainframe computer will cater to a large number of work-stations or terminals that, provided the distance between the two is not excessive, can be permanently cable-linked. Where the distance is considerable, they can be linked via telephone modem devices. This also makes it possible to cross international frontiers and even link continents.

Small industrial and commercial organizations can gain access to a mainframe computer, and the required software, on a "time-share" basis. The computer, which may be many miles distant, is accessed via the public telephone network. The facility is available to numerous subscribers and each pays for the actual computing time used; but to this cost it is necessary to add the normal telephone charges which can be considerable.

Subscription to such a system also provides access to a range of back-up services such as the use of the latest software and professional advice regarding its application. Help is also available to solve problems encountered when using the system generally.

In addition, time-share subscribers do not have to concern themselves with maintenance or servicing of the system, as is the case with an in-house installation when a service contract with the manufacturer or supplier would be another costly but essential requirement.

One drawback to time-sharing, assuming that the financial considerations are acceptable, is that access to the computer may not always be conveniently available because too many subscribers are trying to "log on," or connect into the system, at any one time. Another is that the data security may be inadequate if the work concerned is of a sensitive nature.

Many organizations have requirements that do not justify the installation of a mainframe, but at the same time could not be satisfactorily allowed for by time-sharing. They often install their own large to medium-sized systems. The available capacity and computing power will be less extensive than that available from a mainframe computer, but still capable of servicing a very large organization.

A feature of these installations is the permanent linking of work stations, or terminals, to the host computer. When terminals are linked in this way, they are said to be "networked." By this means the complete system may become fully interactive, making it possible for data to be originated and accessed by a number of users. It may be possible for a programmer to utilize data originated by a designer, thus providing a link between design and manufacture, referred to as CAD/CAM (Computer-Aided Design/Computer-Aided Manufacturing), which will be presented subsequently. There may also be direct connections to the machine tools on the shop floor, a facility referred to as Direct Numerical Control (DNC).

A variation of this approach is to transfer the completed part program to a temporary data storage facility on the shop floor, which is situated alongside and connected to the machine that is to manufacture the part. The production controller can download the data into the machine control unit as and when required. Programs can also be downloaded from the machine into the storage unit. The computer systems can also be configured to deliver program data to tape punch, cassette tape, or floppy disk units.

The networking of user terminals described above is also available on even smaller computer installations, but the number of terminals and, of course, the computing power available is proportionally reduced. However, a "mini" or "supermini" computer, small enough to fit under a desk, is still capable of handling a comprehensive 3D CAD/CAM system, providing control of both the design and manufacturing elements of engineering and often including other functions such as costing, invoicing, etc.

Finally, there are the installations involving "micro" or "supermicro" computers. Stand-alone CAPP systems that operate on microcomputers are now capable of handling very complex programming requirements and are widely used. This type of installation is relatively cheap, which makes their installation by smaller companies a feasible proposition.

Micro-based systems can be networked and linked to peripheral equipment such as plotters, printers, and tape punches. Provided the distance is not too great, they can also be cable-linked to machine tools.

Because of restrictions on expenditure, most educational establishments have found it necessary to install microcomputer-based CAPP systems, and it is this type of installation which students are most likely to use when first being introduced to computer-aided part programming.

Figure 10.1 shows a general arrangement of a CAPP work station utilizing a microcomputer.

It is possible that a company may not become involved in any of the arrangements outlined above, since there are many consulting offices that offer a program preparation and proving service. The use of a consultant can sometimes be attractive to companies that do not have sufficient programming work to sustain a part programmer working full time, and which may prefer to retain a lower, and therefore cheaper, level of skill on the shop floor. Even when the skill levels on the shop floor are such that part programming could satisfactorily be undertaken, the use of a consultant, providing an already proved program, means that no time is lost between ending one production run and beginning the next.

The staff of programming consultant offices will work in close cooperation with the client company, so that the machining is processed in a way acceptable to all concerned. In some cases they also offer supporting services associated with the selection of tooling and the design of special work-holding arrangements if required.

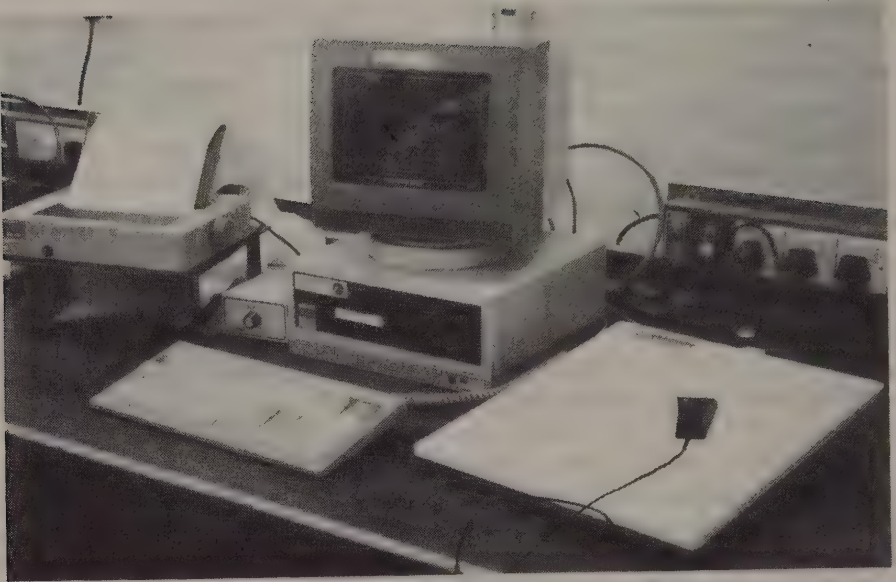


Figure 10.1 General arrangement of a CAPP work station.

HARDWARE CONFIGURATIONS

Whatever the computer installation used, it is possible to establish, in a general way, the hardware configurations associated with CAPP.

The diagram 10.2 shows a very basic computer-aided part programming system: a computer, a data recording facility in the form of a tape punch, a data transfer facility in the form of a tape reader attached to the machine tool, and the machine tool itself.

In Figure 10.3 this basic arrangement is extended and includes, instead of the tape punch and tape reader, a direct cable link to the machine tool, which

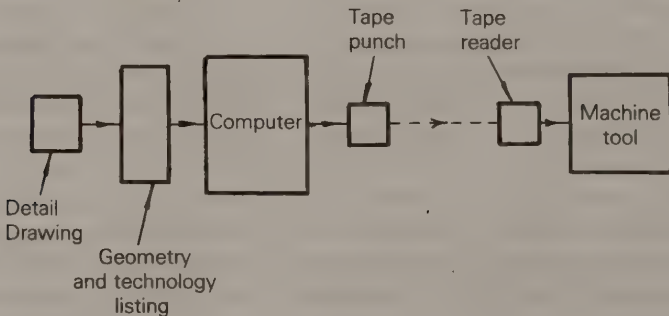


Figure 10.2 Basic CAPP system.

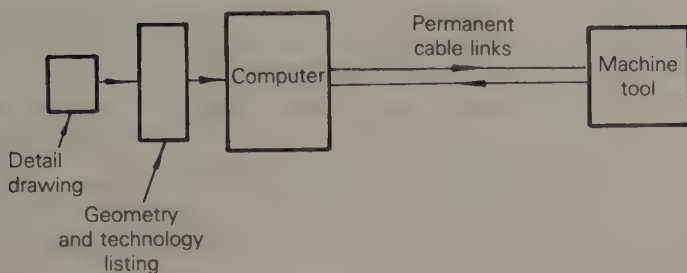


Figure 10.3 Simple direct numerical control.

is a much quicker and more convenient method of transferring data. When the computer and machine are linked by cable, the arrangement is referred to as Direct Numerical Control (DNC).

Figure 10.4 shows the concept extended still further, this time to include cable links to a number of machine tools which may be of different types. These machines may be arranged in a particular way, possibly associated with robot work-handling devices, to form a machining cell.

Taking an even broader view, there may even be a number of programming stations and a number of machines, or machining cells, all linked to the same computer. In this situation the part programming function could be but a small element of a totally integrated computer-controlled manufacturing environment including design, marketing, accounting, materials handling, personnel control, and so on. Complex systems such as this are referred to as Computer Integrated Manufacturing, CIM. Of course, the greater the demands on the system the more extensive will be the required computing and data storage facilities.

From the foregoing the reader will appreciate that it is possible for the CAPP system with which he or she is concerned to be a relatively simple stand-alone

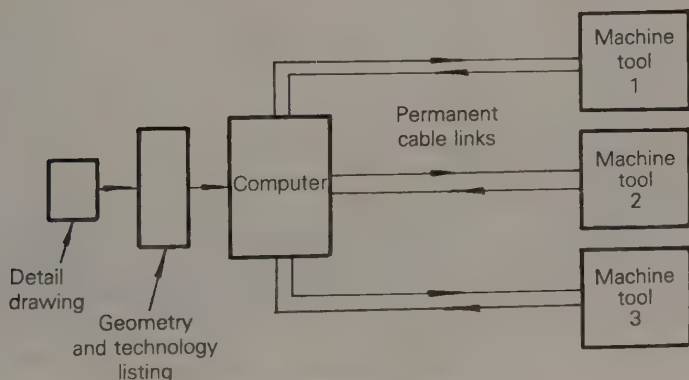


Figure 10.4 Direct numerical control of a machine group.

system, the purpose of which is solely to produce CNC part-programs for a particular machine or a limited number of machines, or it may be an integral part of a much more complex installation.

Even if the overall computing arrangement is complex, it is still possible to consider the CAPP element in isolation and return to the basic objective: preparing a part program with computer assistance and then transferring the resulting data to the machine tool. The main elements in this process are shown in Figure 10.5. Also indicated is the range of peripheral equipment that can be used to support the activity.

The CAPP system available for use by the reader may include all or only some of the items indicated. Systems are structured according to the funds available at the time of purchase, and limited finance can have a restricting effect.

However, it is essential that a part programmer is fully conversant with any system he or she is going to use. Time spent in getting to understand the system

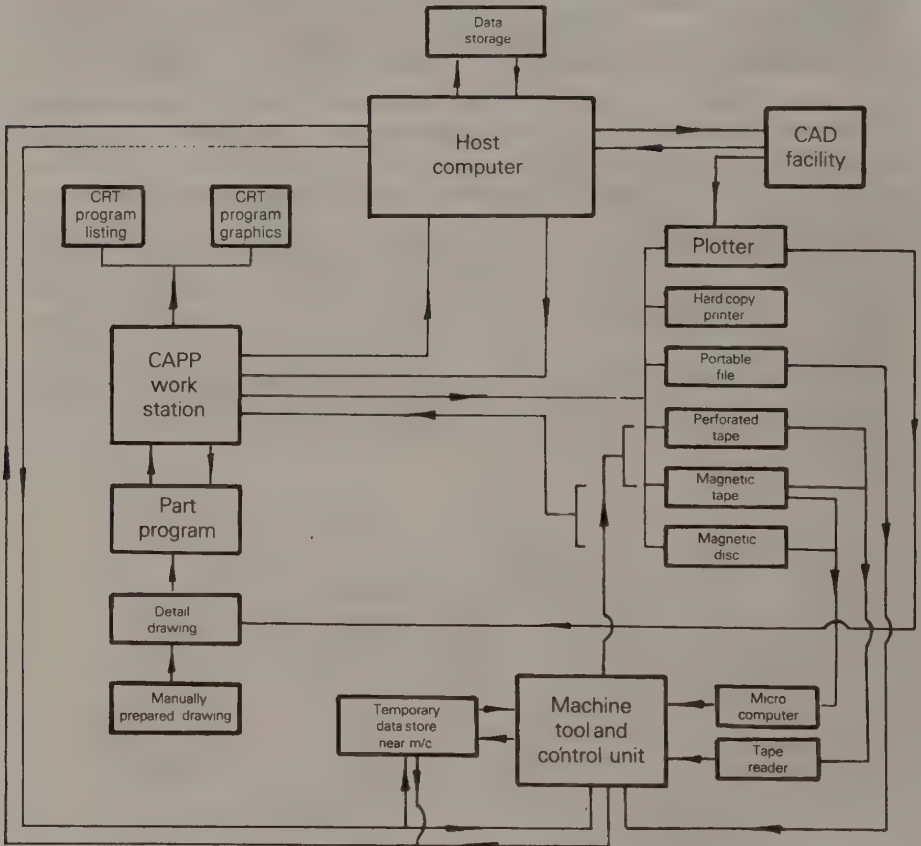


Figure 10.5 CAPP station incorporated in a system.

before making any attempt to prepare a part program is time well spent, since it can do much to eliminate the time-consuming and often frustrating need to ask for help every time an unfamiliar feature of the system is encountered.

INPUT AND CONTROL DEVICES

There are a number of ways by which data can be entered into a computer during the CAPP process.

Input via the familiar alphanumeric keyboard is laborious and rather slow, bearing in mind that the average programmer is likely to be a little lacking in keyboard skills. But since many of the data entries associated with CAPP are repetitive, it is possible to speed up the process by using various supplementary devices and techniques.

Selection from a menu, that is, a list of options, is one such facility. The menu may be included as an overlay on a digitizing tablet or be displayed on the CRT screen.

A digitizing tablet (Figure 10.6a) is a device like a small rectangular board

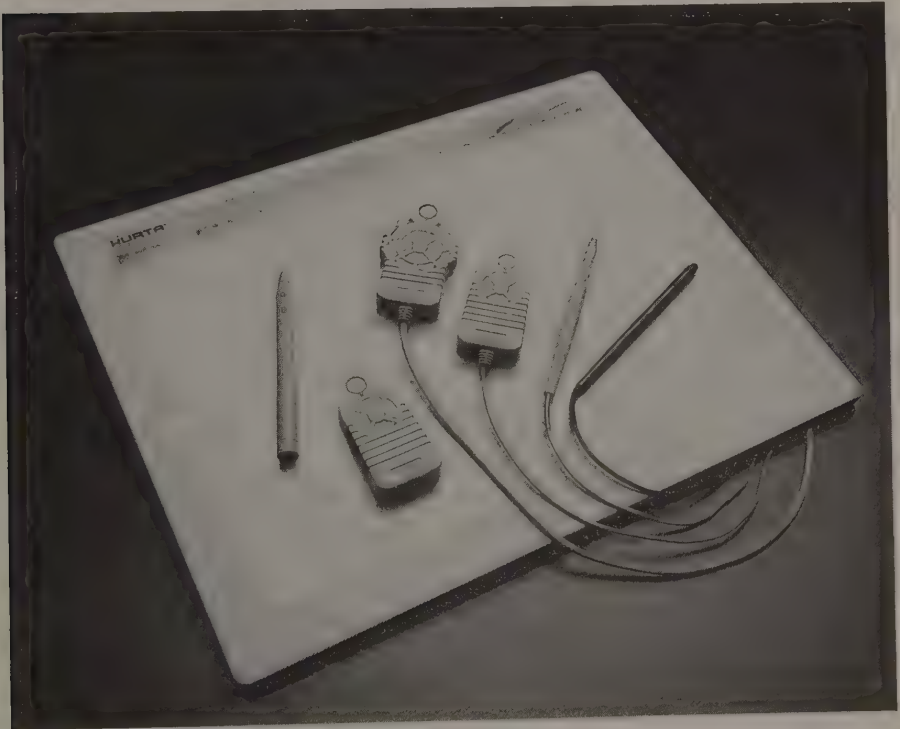


Figure 10.6a Digitizing tablet. (Photograph of IS/ONE courtesy of Kurta Corp., 3007 E. Chambers, Phoenix, AZ 85040.).

that is positioned alongside the computer keyboard and is cable-linked to the computer. The tablet is capable of detecting the position of a puck, light pen, or stylus when any of these devices is placed on it.

An overlay is like a plan that divides the surface of the tablet into a number of small areas. Each area is allocated to a specific function such as representing an item from the menu.

A puck is a device with cross-hairs mounted in a small block. It is traversed by hand over the tablet until the cross-hairs are located over the function to be selected. The selection is then confirmed either by an appropriate keystroke or by pressing a button on the puck. A light pen or stylus is rather like a pen. It is used to identify the menu item required and the choice is then activated by pushing a button or applying slight pressure to the tablet or CRT screen position.

A digitizing tablet is a feature of the programming station illustrated in Figure 10.1, which also shows the puck used for menu selection.

Screen menus can either occupy a complete CRT screen, in which case the graphic image which is an important feature of the CAPP process is temporarily lost, or it can occupy part of the screen so that the graphic image is retained. On small CRTs, the second arrangement can mean that the graphic display is rather cramped. A hardware configuration that will eliminate the disadvantages inherent in both arrangements is to have a two-screen display, one for the menu and program listing and one for the graphic display, but this does add to the cost of the installation.

Selection from screen menus can be achieved in several ways. If the menu items are numbered, selection may be via a keyboard input. A second method is to use a light pen, which involves directing a light source at the CRT screen that is of a special type known as a "vector refresh" screen. A variation of this approach is a pen that senses the light being emitted from the screen itself.

Third, the menu items can be selected by moving a cursor—a spot or cross that can be moved about the screen—and then activating the function by making a keystroke or by pressing a button on the cursor control device.

There are a number of devices that are used to control cursor movement. One of the most commonly used is the "mouse." It has some resemblance to a real mouse because of its shape and its long tail which is, in fact, the cable connecting it to the computer. Beneath the mouse is a set of wheels/balls; as the mouse is moved about a flat surface alongside the computer—a table top, for instance—the wheels/balls detect the movement and cause the screen cursor to make a corresponding movement.

Another device is the "tracker ball." This device has a partially exposed ball mounted in a small box. The ball is rolled around by the palm of the hand. The movement of the ball is detected and, as with the mouse, a corresponding cursor movement appears on the screen.

A third method of cursor control is by the use of a "joystick." As this is

moved around in all directions, the screen cursor moves in unison. A joystick is shown in the illustration of the CAPP work station in Figure 10.6b.

It is possible that an area of a digitizing tablet can be allocated to represent the CRT screen. The stylus or the puck referred to earlier can then be used to pass over the screen area and effect a corresponding movement on the screen.

Apart from menu selection, a cursor is also used within the CAPP process to identify geometric elements such as points, lines and circles that have been constructed on the screen. For example, to construct a line at 90° to a base line will first need a menu selection to identify the type of construction required, followed by identification of the point on the base line at which the second line is to be constructed. The cursor will be moved to identify the point, and the function activated by making a keystroke or pressing a button on the cursor control device. Similarly, a line may need to be deleted, in which case it is first identified by positioning the cursor and then removed by making a keystroke or by pressing a delete button on the cursor control.

The foregoing description has, of necessity, been general in nature. Cursor



Figure 10.6b Part programming station showing use of computer graphics for program proving.

controllers, even of one particular type, are very variable in design. The way in which cursors are used varies from one CAPP system to another. The essential thing to appreciate is that a part programmer is not required to be familiar with all the possibilities; nevertheless, it will be necessary to become competent in the application of the particular devices associated with the program system he or she will be using, and this can only be achieved by using the equipment.

COMPUTER-ASSISTED PROGRAMMING ACTIVITIES

In Chapter 8 the procedure for manual part programming, taking the detail drawing as a starting point, was listed. That list is reproduced below, but this time the programming activities that will be assisted by the use of CAPP are shown in bold type.

1. Select a machine capable of handling the required work.
2. Prepare a schedule of machining operations.
3. Determine work holding and location techniques.
4. Determine tooling requirements and their identity.
5. Document, or otherwise record, instructions relating to work holding, work location, and tooling.
6. **Calculate suitable cutting speeds and feedrates.**
7. **Calculate profile intersecting points, arc centers, etc.**
8. **Determine appropriate tool paths including the use of canned cycles and subroutines.**
9. **Prepare the part program.**
10. **Prove the part program and edit as necessary.**
11. **Record the part program for future use.**

Before proceeding further it is necessary to make the point that the more practical elements of the list, that is, those numbered from 1 to 5, are as necessary for CAPP as they are for manual part programming. CAPP does not eliminate the need for the programmer to have a good working knowledge in the practicalities of metal cutting. The reader is referred to the fuller consideration given to these aspects of CNC machining which is included in previous chapters.

To return to the CAPP process. It will be assumed that consideration has been duly given to the practicalities of a particular machining task, and the computer-aided element of preparing the part program can begin.

It is not possible to list precisely the stages in the CAPP process since there is some variation in approach according to the type of system being used. But in general terms the stages may be itemized as follows:

1. Define the geometric detail of the component. This will involve a series of individually constructed elements that embraces the final component detail.
2. Use the geometric detail to define appropriate machining sequences.
3. Supplement the proposed machining sequences with technology data relating to tooling, feedrates, spindle speeds, etc.
4. Process these data to determine tool paths and to produce a cutter location data file.
5. Produce a CRT screen or paper tool path plot for initial program prove out.
6. Postprocess these data into a form or language that is acceptable to the machine to be used.
7. Transmit these data direct to the machine tool. Alternatively, a punched tape may be produced, or the program otherwise recorded for future use.

CAPP SYSTEMS

Before a CAPP system can be used to prepare a program, time will have to be spent in becoming familiar with the techniques or language to be used—just as it is necessary to study the language of a machine control system before programs for a particular machine can be prepared manually. But the use of CAPP does have a major advantage in this respect: it is probable that the programmer will be required to become familiar with only one technique, since it is possible to postprocess or translate the data into whatever machine control language is to be used.

It is not possible in a text of this nature to give a comprehensive review of every CAPP system, since there are far too many currently in use. Neither would it be of value to consider a particular system in detail. Later in the text, however, programming examples are included in an attempt to give at least a general impression of some of the variations that exist. These examples will be given in the Compact II language by Manufacturing Data Systems Incorporated and APT (Automatically Programmed Tools). Compact II and APT were selected as examples for their leading roles in the development of CAPP and their major influences in CAD/CAM evolution.

In reality, a part programmer will find it necessary to devote as much time as possible to becoming proficient in the application of the particular system he or she will be required to use. Some of the techniques and skills developed in the use of one system are likely to be transferable to another if the need arises.

Although the number of CAPP systems available are many and varied they may be generally defined as being either language or graphics based.

The basic difference between the two concepts is the way in which the ap-

propriate tool paths for the machining sequences are ascertained. The following text deals with language-based programming. Graphics-based programming will be discussed further.

LANGUAGE-BASED SYSTEMS

Early CAPP systems were entirely language-based, the geometry of the part being described by a series of statements constructed from letters of the alphabet, numbers, and a few other symbols. The systems were not interactive, there was no indication if errors had been made in the data, and therefore no correction was possible during the input process. Confirmation of the validity of data could only be ascertained by processing it. If necessary, the program could then be edited. The only visual confirmation of the program data was via a diagram produced on an interfaced plotter after the data entry was completed.

Modern language-based CAPP systems also use alphanumeric input supplemented by certain symbols, but have been considerably improved by the incorporation of computer graphics. As data are entered, there is an instant corresponding graphic display giving an indication of the validity of the input. The systems are fully interactive: if data are not acceptable, the fact will be indicated, often with messages to indicate why this is so, and the programmer can then act on this information and modify the input as programming proceeds.

Having defined the component by the use of a series of geometric statements, the programmer then selects elements from the overall construction and includes these in what is, in effect, a composite statement that will form the basis of a particular machining operation. The composite statement, which may represent, for example, a profile or a series of holes, will be expressed in language form.

The composite statements are now supplemented by data relating to speeds and feeds, tooling, etc. These are referred to as tool change data statements, and will be discussed further.

At this stage these data are processed to determine tool paths and to generate a cutter location data file, referred to as CL Data. Finally, the data are post-processed to generate a program in machine tool code for the particular machine to be used. Each of these stages is explained further in the following text.

GEOMETRIC DEFINITION

It is assumed that a person making a study of CAPP will already be familiar with manual part programming techniques, and will therefore appreciate that

any machined feature or profile can be geometrically defined. He or she will already be familiar with defining tool movements in relation to the workpiece as being linear, circular, and point-to-point. An appreciation of how one geometric feature can intersect with another and the need to define dimensionally such intersection points should also be well understood.

In manual part programming the profile is, in effect, split up into its geometric elements. This is also the case with CAPP, in which the shape or feature to be machined is expressed in terms of directions, distances, lines, points, and circles.

The way in which these geometric elements are generally defined when using CAPP systems is listed below. The lists should not be considered to be definitive since the approach to geometric construction differs between programming systems, and there are also variations in the words used to describe what is essentially the same feature. A further complication is that some of the definitions used, while perfectly logical and therefore acceptable when applied to geometric construction involving computer graphics, do not conform to true mathematical expression. However, the reader is assured that the descriptions that follow are typical of those likely to be encountered.

Directions are defined in the usual way, that is by the use of the letters X, Y and Z which relate to the axes of movement of the machine tools. Distances are given a dimensional value in millimeters or inches and angles are stated in degrees.

A point may be defined in a number of ways as follows:

- (a) As a zero;
- (b) as a point with known Cartesian coordinates;
- (c) as a point with known polar coordinates;
- (d) as an intersection of two straight lines;
- (e) as an intersection of a straight line and a curve;
- (f) as an intersection of two curves.

Examples of the above definitions are illustrated in Figure 10.7 (a) to (f), respectively.

Straight lines may also be defined in a number of ways and the following descriptions correspond to the illustrations in Figures 10.8 (a) to (h).

- (a) As being parallel to a stated axis;
- (b) as being at a known distance from a previously defined point and at a known angle to a previously defined straight line;
- (c) as being between two known points;
- (d) as being tangential to two known circles;
- (e) as being parallel to a defined straight line;
- (f) as being perpendicular to a defined straight line;
- (g) as being perpendicular to a defined point;
- (h) as passing through a defined point, at an angle.

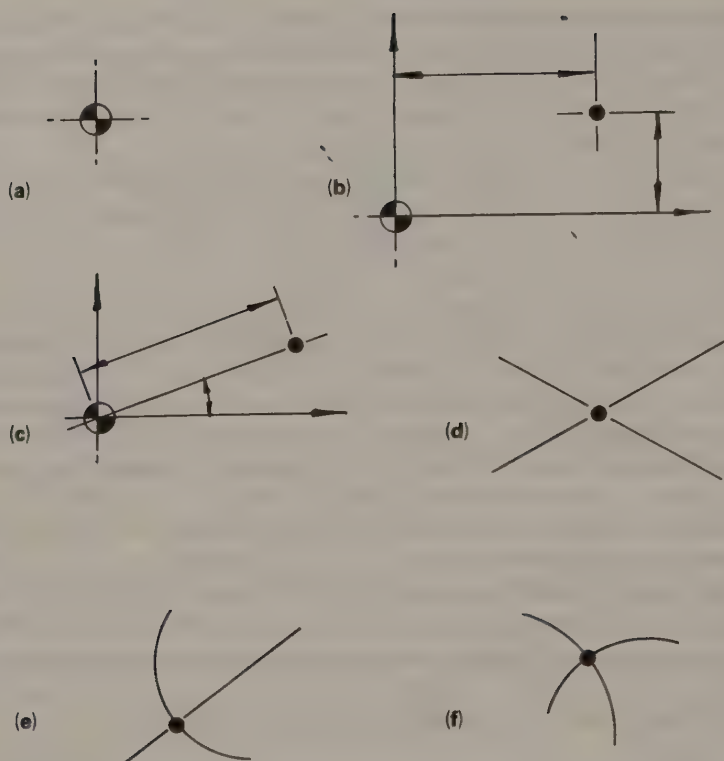


Figure 10.7 Point definition.

Circles may be defined as follows, and as illustrated in Figures 10.9 (a) to (g).

- (a) As a known radius passing through two defined points;
- (b) as an unknown radius passing through three defined points;
- (c) as a center point and passing through a defined point;
- (d) as a center point and tangential to a defined straight line;
- (e) as being tangential to a defined line, passing through a defined point and with a known radius;
- (f) as being tangential to two defined lines and with a known radius;
- (g) as being tangential to three lines and with a known radius.

A further complication with some constructions is that two versions are sometimes possible. Consider Figure 10.10(a), a radius passing through two defined points. One construction is shown in full line and an alternative construction is shown in broken line. Similarly, the construction shown in Figure 10.10(b), of a circle of given radius tangential to two defined lines, has four possible versions as indicated.

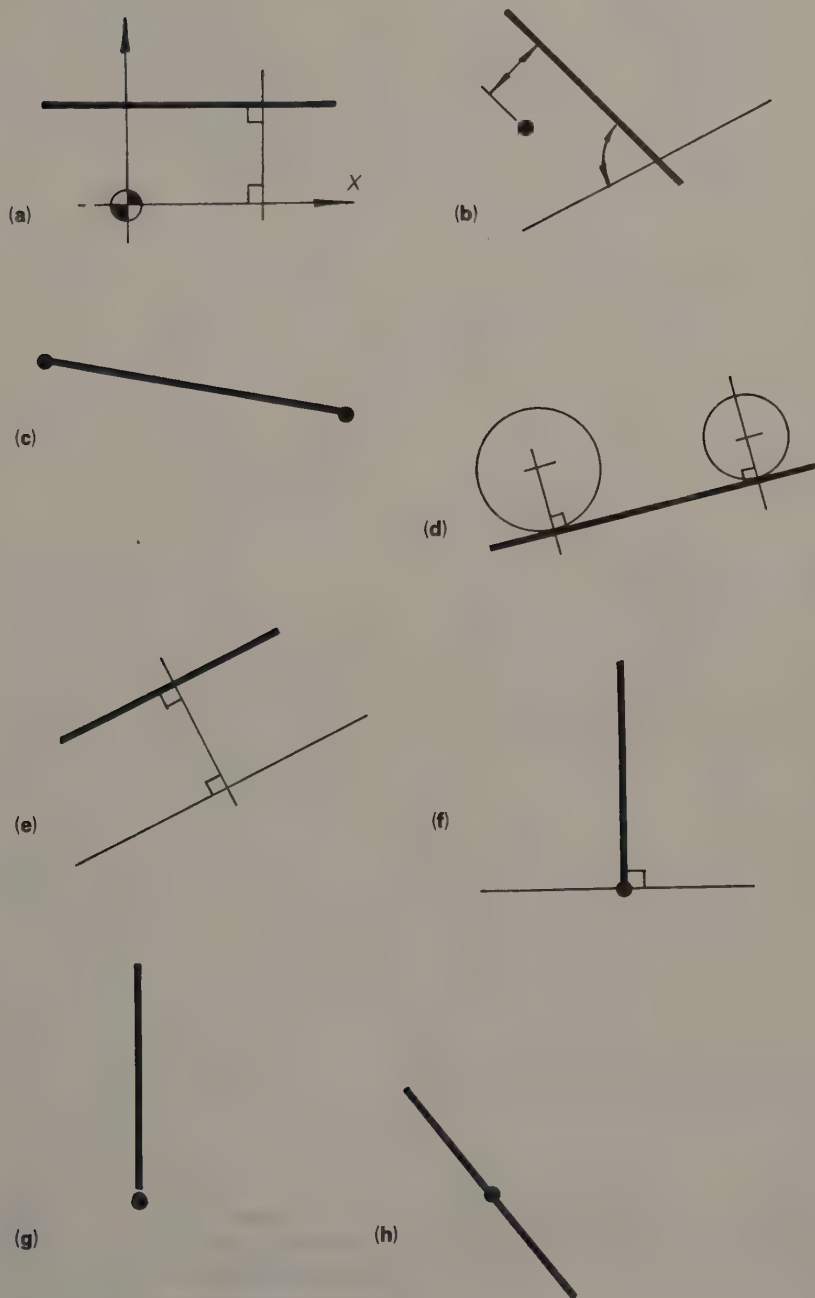


Figure 10.8 Line definition.

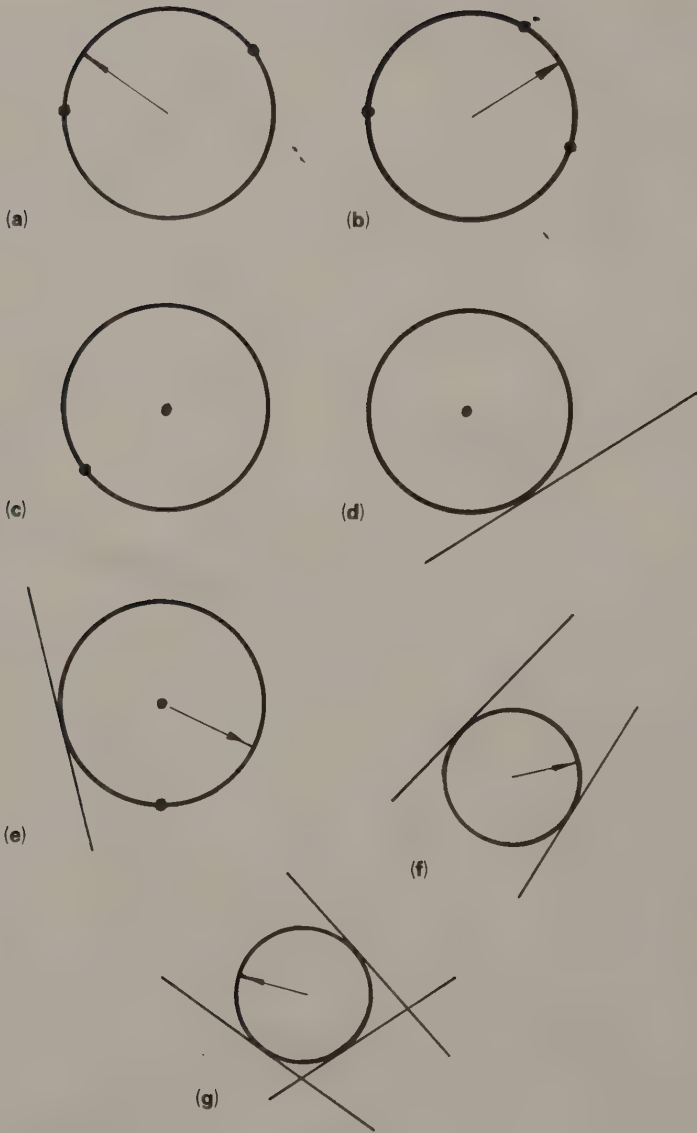


Figure 10.9 *Circle definition.*

Clearly there is a need to clarify the situation by providing the new element with a sense of orientation or direction in relation to the existing geometry. The way this is achieved differs between one system and another and the student will require specific instructions relating to the system he or she will be using.

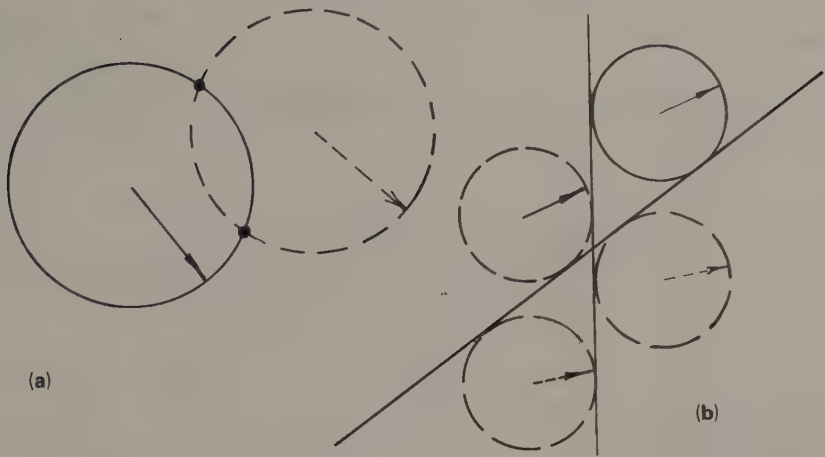


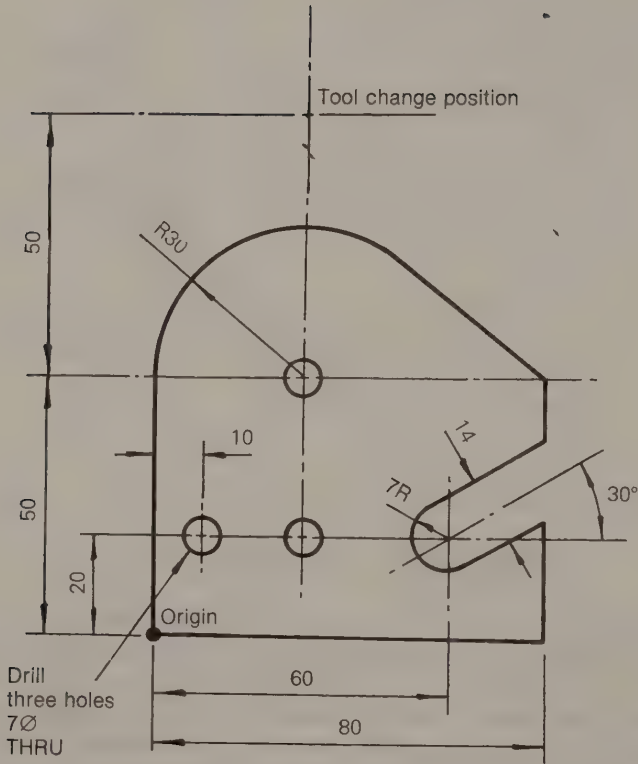
Figure 10.10 *Alternative constructions.*

GEOMETRY FILE CONSTRUCTION

The programmer begins the process of geometric definition by first studying the profile or feature to be machined and then giving each element an identity. He or she may do this by marking a drawing prior to entering data into the computer or, if sufficiently competent in the use of the system, the programmer may allocate identities as the entries are made. The student is likely to benefit, at least in the early stages of using a particular CAPP system, by adopting the first approach. As with all methods of programming a logical approach is essential to avoid making frustrating mistakes.

The precise method used to identify elements varies from one system to another, but it is common practice to give each element a numerical identity, which is then followed by the appropriate definition. Thus a line identified as line number 7 that is to be constructed from point number 1 at an angle of 90° may be programmed simply as L7, P1, A90. A complete profile consisting of a series of lines, circles, and points previously defined may be listed as follows: PF, P1, L1, L2, L3, L4, C1, L5, L6, P2. The initials PF identify the statement as being a profile.

It should be possible to gain a general appreciation of the techniques used by studying the two examples of geometry statements listed below. Both lists relate to the milled component illustrated in Figure 10.11. For the sake of simplicity the problems associated with holding such a component while the profile is machined have been ignored. Normally, clamping arrangements and work-holding devices have to be accommodated within the part program if collisions are to be avoided. Areas they will occupy, which in effect become "no go" areas for the tool, have to be identified dimensionally and may be displayed graphically as part of the general geometry.



Material: mild steel 10 mm thick.

Figure 10.11 Component detail.

The program definition lists which follow were prepared by two different CAPP systems for comparison. The result is two different approaches to the geometry definition. Reference to Figures 10.12 and 10.13 will indicate how each definition refers to the part geometry.

The reader will note the common use of P or PT to indicate a point, L or LN to indicate a line, and C or CIR to indicate a circle. But there, apart from the numerical identity referred to above, the similarities end. Further study of the lists will show that the variations become even more pronounced when the individual elements are compared.

Example No. 1 Software: Compact II (Figure 10.12)

MACHIN, MILL
IDENT
SETUP, 30LX, 100LY, 100LZ
BASE, 0XB, 0YB, 20ZB

Statement identifies machine used
Program identifier
Machine setup information
Absolute zero location

DPT1, 0XB, 0YB, 0ZB
DPT2, PT1, 10X, 20Y
DPT3, PT2, 30XB
DPT4, PT3, 50YB

DPT2, PT1, 10X, 20Y

DPT3, PT2, 30XB

DPT4, PT3, 50YB

DCIR1, PT4, 30R

DCIR2, 60XB, 20YB, 0ZB, 7R)

DLN1, XB

DLN2, 80XB, 50YB, 0ZB, CIR1, YL

DLN3, 80XB

DLN4, YB

DLN5, 30CCW, CIR2, YL

DLN6, PARLN5, CIR2, YS

Definition of point locations

Definition of circles

Definition of lines

Example No. 2 Software: APT

PARTNO PROFILE DEFINITION EXAMPLE
CUTTER/0
MACHIN/MILL

Program identification

Cutter call up

Statement identifies machine used

P1 = POINT/0,0,20

P2 = POINT/10,20,

P3 = POINT/30,20,

P4 = POINT/30,50,

```
C1 = CIRCLE/CENTER,P4,RADIUS,30 }
```

C2 = CIRCLE/60,20,20,7

```
L1 = LINE/P1,PARLEL,LY
```

```
L2 = LINE/80,50,20,RIGHT,TANTO,C1
```

```
L3 = LINE/PARLEL,L1,XLARGE,80
```

```
14 = LINE/P1,PARLEL,LX
```

```
L5 = LINE/LEFT, TANTO,C2,ATANGL,30,YLARGE
```

```
L6 = LINE/PARLEL,L5,YSMALL,14
```

Definition of point locations

Definition of circles

Definitions of lines

```
SETPT := POINT/30,100,100
```

Machine setup point

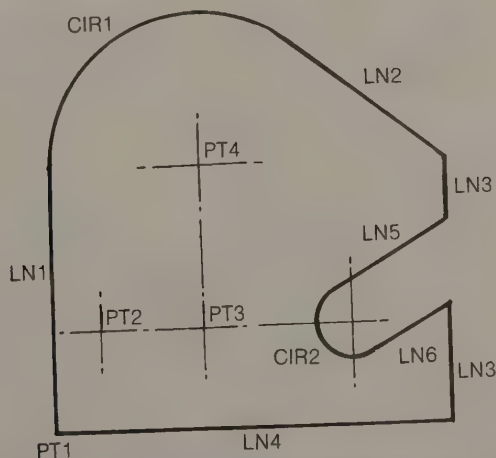


Figure 10.12 Profile definition—MDSI Compact II (Manufacturing Data Systems Inc.).

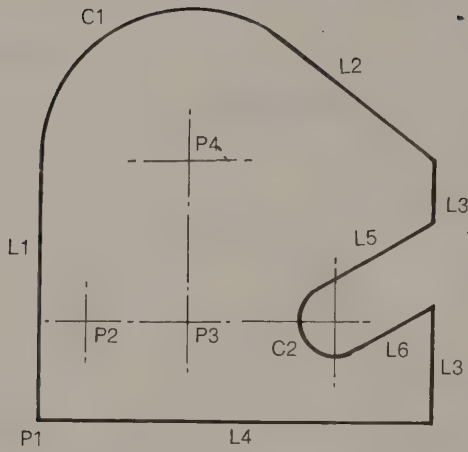


Figure 10.13 Profile definition—APT (Automatic Programmed Toolpath).

Now that we have looked at two examples of geometry statements from popular computer-assisted languages let us review an entire program in Compact II. It is noted that programs of this type can be broken down into five major areas of initialization, geometry, tool change, tool motion, and termination as can be seen in Figures 10.14 through 18. The programming process

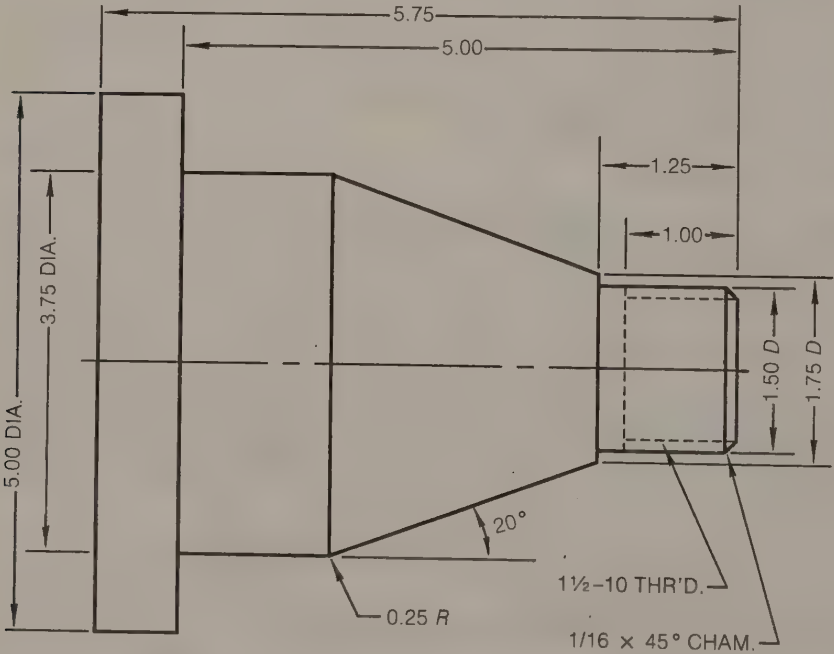


Figure 10.14 Spindle bolt part print. (From Numerical Control Technology Workbook, courtesy of Schlumberger Technologies Inc., Ann Arbor, MI).

starts from a part print such as Figure 10.14, from which the programmer decides on the machine, tools, and process to use. In this example the programmer has decided to use three tools, the first for rough turning leaving 0.020 in. finish stock on the diameter and 0.005 in. stock on the length for finish. The second tool selected is for finish turning, while the third and final tool is for a single-point threading operation.

The second step the programmer has taken is to determine what the machine setup should be for this program. Figure 10.15 shows the machine layout decided upon. The machine zero is located at the chuck end stop end of the part and the machine tool turret reference point will be established by the operator 10.5 in. from this point in the Z axis. The X axis turret reference point will be established 8.0 in. from the center of the workpiece.

Section I Initialization

Once the machine setup has been determined, the program initialization statements can be written for the computer program. Figure 10.16 shows the five

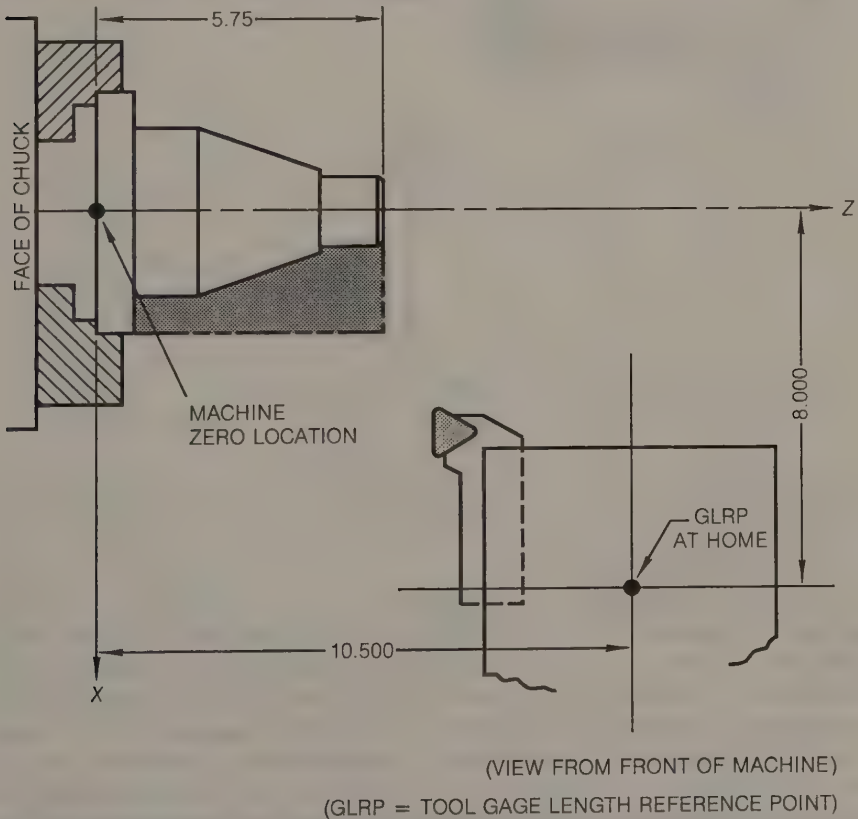


Figure 10.15 Machine setup layout. (From Numerical Control Technology Workbook, courtesy of Schlumberger Technologies Inc., Ann Arbor, MI.).

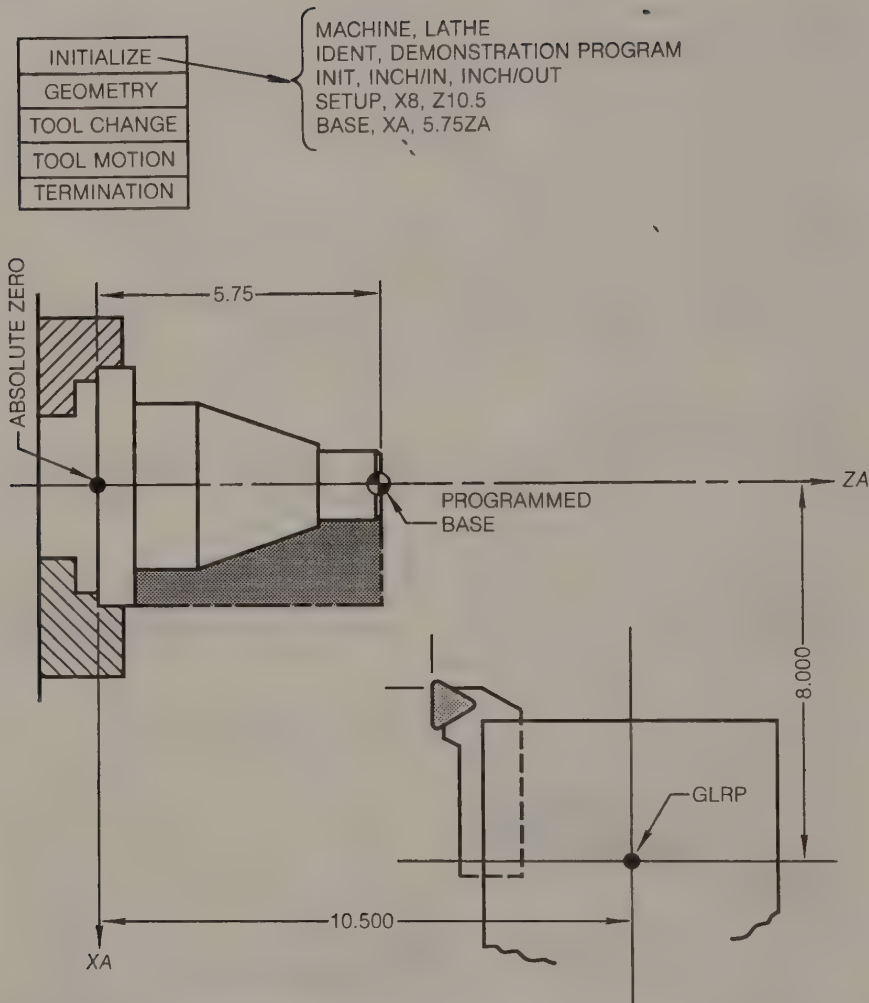


Figure 10.16 Initialization statements. (From *Numerical Control Technology Workbook*, courtesy of Schlumberger Technologies Inc., Ann Arbor, MI.).

initialization statements for a Compact II program. Starting with the MACHINE statement the computer is told what specific type of machine tool it is creating a program for. Since this is a demonstration program, just a generic post or machine file is used. For each specific type of machine and control there will be a special command. Next is the IDENTification statement, which assigns the title to be output on the various files generated by the computer. The third line is a INITialization statement to indicate whether input will be inch or metric and whether output is wanted in inch or metric. The fourth statement SETUP informs the computer as to where the machine tool turret reference point is in

relation to the machine absolute zero point location. The fifth and final initialization statement is BASE. The base statement informs the computer of the location of the program zero location, used for defining part geometry, in relation to the machine absolute zero.

Section II Geometry

The second major section of a computer program involves the description of the part's geometry to be machined. The programmer first marks up the part print as to the geometry required to machine the part. The geometry labels are normally placed on the part as in Figure 10.17 as a record for future reference. In the example in Figure 10.17 the programmer has defined two lines and two

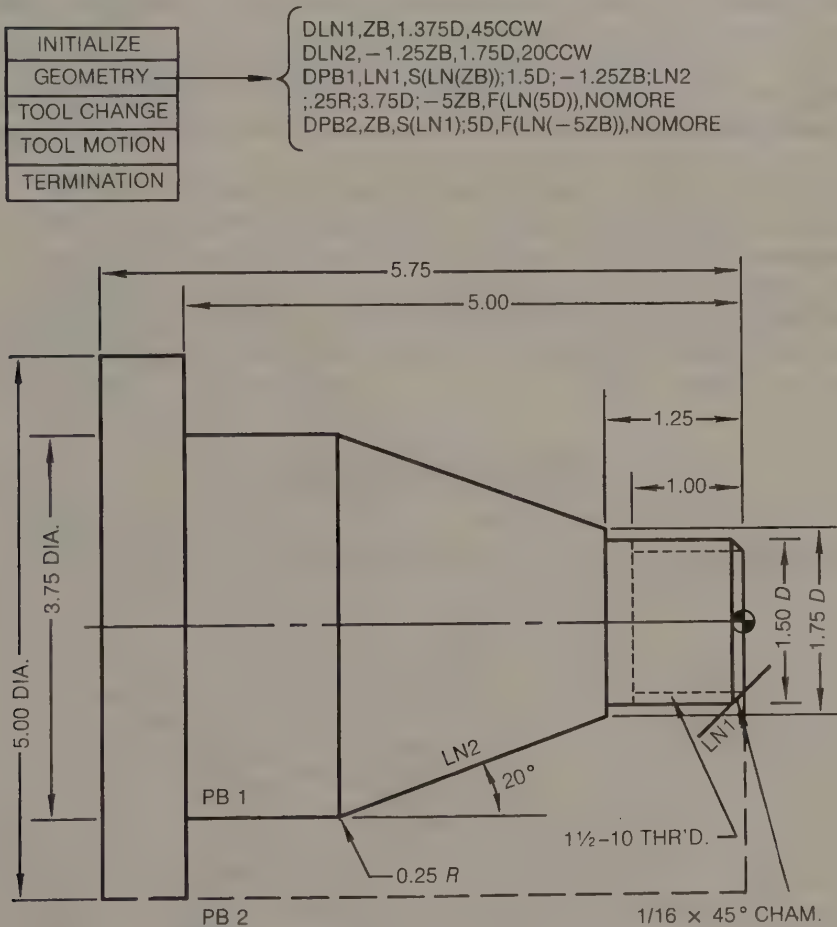


Figure 10.17 Part geometry definition. (From Numerical Control Technology Workbook, courtesy of Schlumberger Technologies Inc., Ann Arbor, MI.).

part boundaries. Line LN1 represents the $1/16 \times 45^\circ$ chamfer, while Line LN2 is the 20° part angle. Part boundary PB1 is defined as the finish part outline, while PB2 is the outline of the rough stock.

The amount of geometric definitions required and their complexity will vary with the skill of the programmer and the capability of the language to handle complex statements. Computer systems will handle geometry verifications in different ways. Some systems process all statements in a batch mode together for statement validity and then allow the system to plot the results on paper or a CRT screen. Other language-based systems are fully interactive: the correctness of data input is verified as programming proceeds and error messages are displayed if appropriate. This ensures that the data input is acceptable to the system, but it does not necessarily ensure that the programmer has not made other mistakes, so it makes sense to reprocess the input in its entirety as a final check.

When the geometry statements have been verified as being correct, it is possible to obtain a print from an interfaced printer of the data listing. This may be required for filing for future reference, forming part of the general documentation relating to that particular job.

It is also possible to obtain printed copies of the graphic construction developed during the programming process. Figure 10.18 illustrates the graphics that appeared on the plotter when, using the Compact II software, the geometry statements were entered for the lathe exercise in Figure 10.17.

From the plot the programmer can easily see the various pieces of geometry defined and can visualize the outline of the rough and finish piece part. Geometry relationships and gross errors can be determined in this type of plot.

TOOL CHANGE DATA

After the geometry of the component has been defined, the programmer has to consider the more practical aspects of producing a machined component,

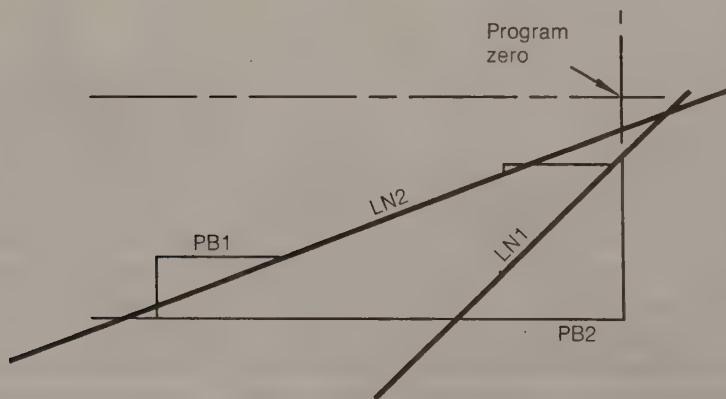


Figure 10.18 Geometry plot.

such as the sequence of operations, the cutting tools to be used, and the choice of appropriate cutting speeds and feeds. If the programming task has been approached in a logical manner, most of these aspects will have been considered before the CAPP process was started. Now data defining these factors have to be added to the part program to supplement the geometric data previously entered. It is possible that some computer assistance may be available.

Tool Change Data statements are included in the Compact II and APT program examples that follow. See Figure 10.19. The tool change statement will be found in various locations of the machining body of the program, wherever a tool selection is required, and will be followed by its tool motion statements.

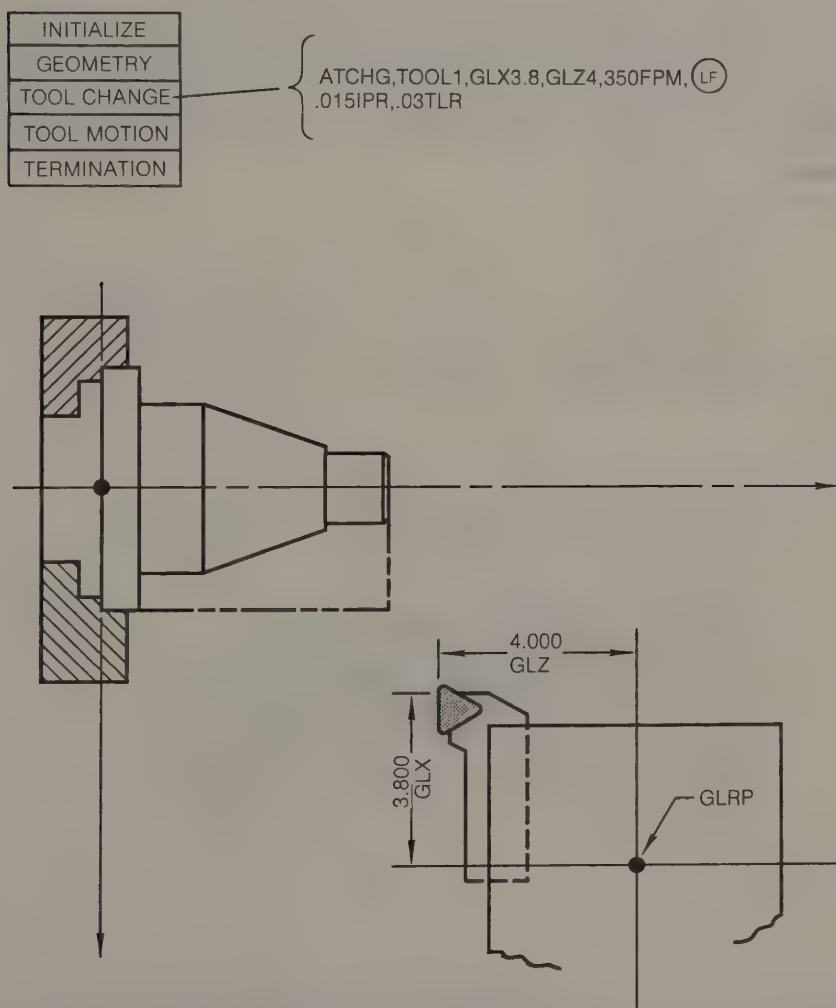
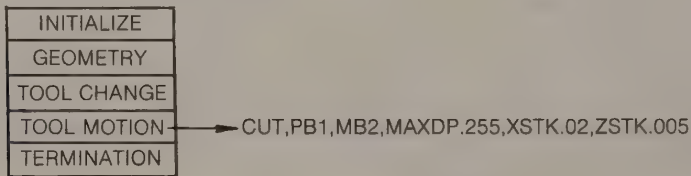


Figure 10.19 Tool change data. (From Numerical Control Technology Handbook, courtesy of Schlumberger Technologies Inc., Ann Arbor, MI.).

Tool statements normally will include information on tool number, gage lengths and diameters, cutting speeds, and feed rates.

On some systems the tooling available for a particular machine will be listed and contained on file within the programming system. Such a data file is referred to as a “tool library” and can be displayed on the screen. All the data relevant to a particular tool, such as the material from which it is made, its shape, and its dimensions, will be indicated together with an identity code for use within the part program. The dimensions of the tool, in particular its radius or diameter, are of particular importance since they will have a direct effect on the cutter paths automatically generated at the next stage of the CAPP process.

Cutting speeds and feeds can be determined without computer assistance, and entered into the program in much the same way as when preparing a manual CNC part program. On the other hand, there may be assistance via the system in response to data input identifying the cutting tool and the part material. The correct speeds and feeds will then be determined automatically and included in the program.



(One statement will produce all the roughing passes indicated.)
This statement causes the rough cutting of material between part boundary 1 and material boundary 2. A finish stock of 0.02 in. will be left in the X axis, while 0.005 in. will be left in the Z axis.

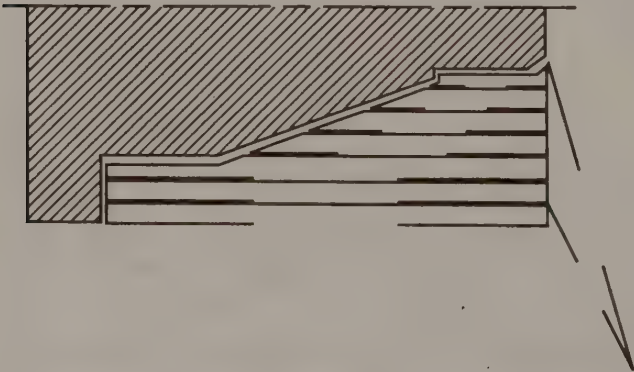
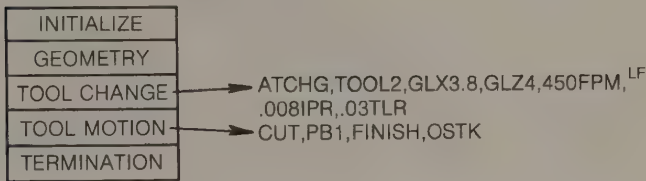


Figure 10.20 Roughing cutting cycle in Compact II. (From Numerical Control Technology Workbook, courtesy of Schlumberger Technologies Inc., Ann Arbor, MI.).

TOOL MOTION DATA

The fourth major section of a CAPP program is Tool Motion statements, which are grouped with the corresponding tool change statement. These statements will determine how the part will be cut. Machining cycles and feed/rapid traverse moves will be established. Knowledge of the particular language used is important here because there are many options from which to choose. Figures 10.20, 10.21, and 10.22 give only a few examples of tool motion statements for lathe work.

When defining the sequence and types of machining operations to be carried out, the programmer will be required to take into consideration the special cycles that are an inbuilt feature of the programming system. All the normal machining sequences—drilling, screw cutting, face milling, boring, etc.—are likely to be catered for. It will also be possible to generate subroutines. To use these facilities effectively the programmer will need to be fully conversant with the particular system being used, and this is only achieved by experience.



The first statement calls for a tool change to the finish turning tool.
The second statement calls for the finish profile of the part to be cut.



Figure 10.21 Finishing turning—Compact II. (From Numerical Control Technology Workbook, courtesy of Schlumberger Technologies Inc., Ann Arbor, MI.)

INITIALIZE	
GEOMETRY	
TOOL CHANGE	→ ATCHG, TOOL3, GLX3.8, GLZ4, TLR0, ^{LF}
TOOL MOTION	→ 50FPM, .01IPR
TERMINATION	→ THRD, CX, LEAD(1/10), .2ZB, -1.2LZ, MAD1.5, ^{LF}
	→ MID1.3773, SDPTH.02, FDPTH.005, 30DEG
	→ END

The first statement calls for a tool change to the threading tool. The second statement calls for a multiple threading cycle to be generated. The third statement calls for termination of the computer program and commands the tool back to its home position.

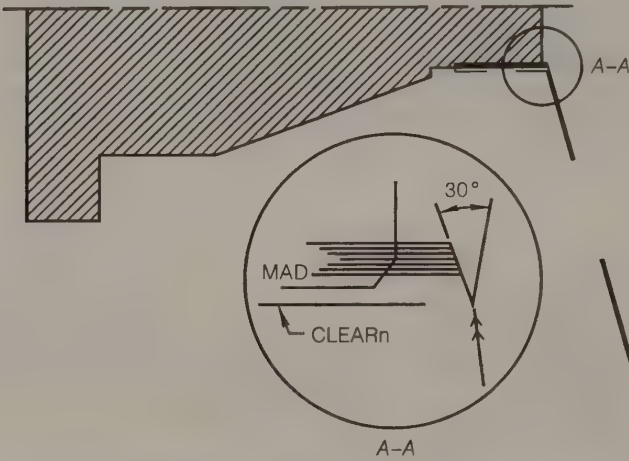


Figure 10.22 Threading cycle—Compact II. (From *Numerical Control Technology Workbook*, courtesy of Schlumberger Technologies Inc., Ann Arbor, MI.).

TERMINATION

The fifth major section of most CAPP programs is the termination of the program. Many systems use a single statement of END or FINISH to indicate that the program is complete. This single statement will normally cause the generation of machine "CNC" program steps to return the tool to its home position and complete an orderly machine shut down. The complete Compact II program can be reviewed in Figure 10.23.

CUTTER LOCATION DATA

When the computer is programmed with all the data defining the part geometry, the machining sequence, tooling, etc., the process of determining the cutter location data, referred to as the CL Data, can begin. In simple terms, the CL

Compact II Part Program

MACHIN,LATHE	Identifies machine used
IDENT,DEMONSTRATION PROGRAM	Program file identification
INIT,INCH/IN,INCH/OUT	Set up data format in and out
SETUP,X8,Z10.5	Machine setup
BASE,XA,5.75ZA	Program zero location
DLN1,ZB,1.375D,45CCW	Define Line 1
DLN2,-1.25ZB,1.75D,20CCW	Define Line 2
DPB1,LN1,S(LN(ZB));1.5D;-1.25ZB;LN2;.25R; 3.75D;-5ZB,F(LN(5D)),NOMORE	Define finish part boundary
DPB2,ZB,S(LN1);5D,F(LN(-5ZB)), NOMORE	Define material boundary
ATCHG,TOOL1,GLX3.8,GLZ4,350FPM, .015IPR,.03TLR	Define roughing tool
CUT,PB1,MB2,MAXDP.255,XSTK.02, ZSTK.005	Create roughing cuts
ATCHG,TOOL2,GLX3.8,GLZ4,450FPM, .008IPR,.03TLR	Define finish turning tool
CUT,PB1,FINISH,0STK	Create finish turning pass
ATCHG,TOOL3,GLX3.8,GLZ4,TLR0, 50FPM,.01IPR	Define threading tool
THRD,CX,LEAD(1/10),.2ZB,-1.2LZ,MAD1.5, MID1.3773,SDPTH.02,FDPTH.005,30DEG	Create threading pass
END	Terminate program

Figure 10.23 Compact II program. (From Numerical Control Technology Workbook, courtesy of Schlumberger Technologies Inc., Ann Arbor, MI.).

Data can be described as the dimensional definition of the cutter path from a defined datum point.

In determining the CL Data the computer automatically calculates the movements necessary to achieve the geometric features previously defined. In doing so, account will be taken of cutter sizes; where appropriate, compensation for the cutter radius will be made. Where area clearance is required, and excess stock material has to be removed, the computer will determine the appropriate tool paths.

CL Data can be viewed on some CRT screens and a printout can be obtained if required.

Tool paths can be displayed graphically, in some cases with a three-dimensional or pictorial effect. It is also possible to produce, via a plotter or a printer, a diagrammatic representation of the part geometry and the cutter paths in relation to that geometry. Both the graphical display on the CRT screen and the plotter output are usually enhanced by the use of different colors to indicate different features: one color for the geometric shape of the component and a different color indicating the paths of each tool to be used, for instance.

In Figures 10.20 through 10.22 the printouts of the cutter paths for programming the component shown in Figure 10.14 is illustrated.

When the CL Data are considered to be correct, the final stage in the CAPP process, that of postprocessing, can be undertaken.

With the CL Data file compiled, it is also possible, on some systems, to determine the time that will be taken to machine the part. The computer calculation is based on the cutting speeds and feeds entered as part of the technology data.

It should be noted that computer languages may be interactive or batch processed. If interactive, the system may check each individual statement for validity, then create its CL Data, postprocess it to machine language, and plot/display the results or error before proceeding to the next statement. Batch processing systems, on the other hand, takes the entire program through each step of processing (statement validity, CL Data, postprocessing, and display) before going on to the next step. In batch processing the computer processing is normally stopped at the end of the step reporting errors with an error statement print out. Interactive program systems will allow error correction "on the fly."

POSTPROCESSING

Postprocessing is the stage in the CAPP process where the CL Data and other information relevant to the machining of the component is assembled into a form that will be accepted and meaningful to the control system of the particular machine to be used. Features such as G and M codes, previously not part of the program data, are now automatically incorporated.

Because there are many variations in the control systems fitted to machine tools, it is necessary to have a postprocessor to suit each control system for which part programs are to be prepared. The manufacturers of CAPP systems will supply specific postprocessors to order, these being immediately available for the more widely used machine controls. It is also possible with some systems to purchase a "writing kit/generic post," which permits users to compile their own postprocessor, for it is actually a relatively simple computer program. In this way, if a new machine is acquired, it can be readily assimilated into the CAPP system.

The postprocessing of data into a machine control language is achieved very rapidly, being simply a case of making a few key strokes. The resulting program can now be recorded for future use in whatever form is deemed to be appropriate.

Examples of postprocessed programs follow in Figures 10.24 and 10.25. Figure 10.24 shows a combination program output that is good for programmer use during machine tryout. This output shows the computer statement followed

by the CL Data or the machine information it generates. It is good for tryout because when an error is found in the machine program you know right away what computer statement generated it. Figure 10.25, on the other hand, is a machine tool operator's program read out. On the machine tool program output no other information is included to cloud the issue.

>MACHIN,LATHE MAIN 9.27 LINK 3.31 SYS 8.23 L# 136	}	INITIALIZATION
>IDENT, DEMONSTRATION PROGRAM 12-22-90 10:55		
>INIT, INCH/IN, INCH/OUT		
>SETUP,X8,Z10.5		
>BASE,XA,5.75ZA = X . Z 5.75	}	GEOMETRY
>DLN1,ZB,1.375D,45CCW = X .6875 Z 5.75 A 45.		
>DLN2,-1.25ZB,1.75D,20CCW = X .875 Z 4.5 A 20.		
>DPB1, LN1,S(LN(ZB));		
>;1.5D;		
>;-1.25ZB;		
>;LN2;		
>;.25R;		
>;3.75D;		
>;-5ZB,F(LN(5D)),NOMORE		
>DPB2,ZB,S(LN1);		TOOL CHANGE Rough Turning Tool
>;5D,F(LN(-5ZB)),NOMORE		
>ATCHG, TOOL1, GLX3.8, GLZ4, 350FPM, .015IPR,.03TLR		

Figure 10.24 Combination file output of part in Figure 10.14. (Note: Programmer-generated statements are indicated by the ">" symbol. All other information is generated by the computer.).

>CUT,PB1,MB2,MAXDP.255,XSTK.02,ZSTK.005
N001 X-01.825 Z-00.6352 F200.00 S0298 T0001 M03
N002 X-00.1 Z-00.0348 F010.00
N003 Z-05.045 F004.47
N004 X 00.255 F004.04 S0269
N005 Z 04.945 F200.00
N006 Z 00.1 F010.00
N007 X-00.41 F200.00 S0336
N008 X-00.1 F010.00
N009 Z-05.045 F005.04
N010 X 00.255 F004.49 S0299
N011 Z 04.945 F200.00
N012 Z 00.1 F010.00
N013 X-00.41 F200.00 S0386
N014 X-00.1 F010.00
N015 Z-03.6317 F005.79
N016 X 00.1428 Z-00.3924 F005.37 S0358
N017 G02 X 00.0172 Z-00.0974 I00.2678
K00.0974 F005.33 S0355
N018 Z-00.9235 F005.30 S0353
N019 X 00.095 F005.07 S0338
N020 Z 04.945 F200.00
N021 Z 00.1 F010.00
N022 X-00.41 F200.00 S0453
N023 X-00.1 F010.00
N024 Z-02.9311 F006.80
N025 X 00.255 Z-00.7006 F005.82 S0388
N026 Z 03.5317 F200.00
N027 Z 00.1 F010.00
N028 X-00.41 F200.00 S0547
N029 X-00.1 F010.00
N030 Z-02.2305 F008.21
N031 X 00.255 Z-00.7006 F006.84 S0456
N032 Z 02.8311 F200.00
N033 Z 00.1 F010.00
N034 X-00.41 F200.00 S0692
N035 X-00.1 F010.00
N036 Z-01.5298 F010.38
N037 X 00.255 Z-00.7006 F008.28 S0552
N038 Z 02.1304 F200.00
N039 Z 00.1 F010.00

TOOL
MOTION
Rough
Turning Tool

Figure 10.24 (Continued)

N040 X-00.41 F200.00 S0941
 N041 X-00.1 F010.00
 N042 Z-00.073 F014.12
 N043 X 00.055 Z-00.055 F013.28 S0885
 N044 Z-01.167 F013.10 S0873
 N045 X 00.1145 F011.52 S0768
 N046 X 00.0855 Z-00.2348 F010.50 S0700
 N047 Z 01.4298 F200.00
 N048 Z 00.1 F010.00
 N049 X-00.2125 F200.00 S1024
 N050 X-00.1 F010.00
 N051 Z-00.0155 F015.36
 N052 X 00.0575 Z-00.0575 F014.33 S0955
 N053 X 03.355 Z 00.7215 F200.00
 N054 X 00.1 Z 00.0215 F010.00
 ABS D 8.4 ZB .75

>ATCHG,TOOL2,GLX3.8,GLZ4,450FPM,
 .008IPR,.03TLR

TOOL CHANGE
 Finish Turning Tool

>CUT,PB1,FINISH,0STK
 N055 X-03.4125 Z-00.6509 F200.00 S1307 T0002
 N056 X-00.1 Z-00.0191 F010.00
 N057 Z-00.0375 F010.46
 N058 X 00.0925 Z-00.0925 F009.29 S1161
 N059 Z-01.17 F009.16 S1145
 N060 X 00.116 F008.03 S1004
 N061 X 00.2754 Z-00.7545 F006.02 S0753
 N062 X 00.3798 Z-01.0447 F004.52 S0565
 N063 X 00.3369 Z-00.9266 F003.72 S0465
 N064 G02 X 00.0169 Z-00.0957 I00.2631
 K00.0957 F003.68 S0460
 N065 Z-00.9285 F003.66 S0458
 N066 X 00.625 F002.76 S0345
 N067 X 01.6408 Z 05.62 F200.00
 N068 X 00.0292 Z 00.1 F010.00
 ABS D 8.4 ZB .75

TOOL MOTION
 Finishing
 Turning Tool

>ATCHG,TOOL3,GLX3.8,GLZ4,TLR0, 50FPM,.01IPR

TOOL
 CHANGE
 Threading Tool

Figure 10.24 (Continued)

>THRD,CX,LEAD(1/10),.2ZB,-1.2LZ,MAD1.5)	}	TOOL MOTION Threading Operation
MID1.3773,SDPTH.02, FDPTH.005, 30DEG		
N069 X-03.4 Z-00.55 F200.00 S0250 T0003		
N070 X-00.0695 Z-00.0401 F002.50		
N071 G33 Z-01.1599 K0.1		
N072 X00.0695 F200.00		
N073 Z 01.2		
N074 X-00.0854 Z-00.0493 F002.50		
N075 G33 Z-01.1507 K0.1		
N076 X 00.0854 F200.00		
N077 Z 01.2		
N078 X-00.0976 Z-00.0564 F002.50		
N079 G33 Z-01.1436 K0.1		
N080 X 00.0976 F200.00		
N081 Z 01.2		
N082 X-00.1063 Z-00.0614 F002.50		
N083 G33 Z-01.1386 K0.1		
N084 X 00.1063 F200.00		
N085 Z 01.2		
N086 X-00.1113 Z-00.0642 F002.50		
N087 G33 Z-01.1358 K0.1		
N088 X 00.1113 F200.00		
N089 Z 01.2		
ABS D 1.6 ZB .2		
>END		PROGRAM TERMINATION
N090 X 03.4 Z 00.55	}	TOOL MOTION Return to Home Position
N091 M05		
ABS D 8.4 ZB .75		
N092 M30		
END MIN: .8 FT: 20.9 MTR: 6.3)		

Figure 10.24 (Continued)

N001 X-01.825 Z-00.6352 F200.00 S0298 T0001 M03	Rough Turning Tool
N002 X-00.1 Z-00.0348 F010.00	
N003 Z-05.045 F004.47	
N004 X 00.255 F004.04 S0269	
N005 Z 04.945 F200.00	
N006 Z 00.1 F010.00	
N007 X-00.41 F200.00 S0336	
N008 X-00.1 F010.00	
N009 Z-05.045 F00	

Figure 10.25 Machine tool output from Compact II program.

N010 X 00.255 F004.49 S0299
N011 Z 04.945 F200.00
N012 Z 00.1 F010.00
N013 X-00.41 F200.00 S0386
N014 X-00.1 F010.00
N015 Z-03.6317 F005.79
N016 X 00.1428 Z-00.3924 F005.37 S0358
N017 G02 X 00.0172 Z-00.0974 I00.2678 K00.0974 F005.33 S0355
N018 Z-00.9235 F005.30 S0353
N019 X 00.095 F005.07 S0338
N020 Z 04.945 F200.00
N021 Z 00.1 F010.00
N022 X-00.41 F200.00 S0453
N023 X-00.1 F010.00
N024 Z-02.9311 F006.80
N025 X 00.255 Z-00.7006 F005.82 S0388
N026 Z 03.5317 F200.00
N027 Z 00.1 F010.00
N028 X-00.41 F200.00 S0547
N029 X-00.1 F010.00
N030 Z-02.2305 F008.21
N031 X 00.255 Z-00.7006 F006.84 S0456
N032 Z 02.8311 F200.00
N033 Z 00.1 F010.00
N034 X-00.41 F200.00 S0692
N035 X-00.1 F010.00
N036 Z-01.5298 F010.38
N037 X 00.255 Z-00.7006 F008.28 S0552
N038 Z 02.1304 F200.00
N039 Z 00.1 F010.00
N040 X-00.41 F200.00 S0941
N041 X-00.1 F010.00
N042 Z-00.073 F014.12
N043 X 00.055 Z-00.055 F013.28 S0885
N044 Z-01.167 F013.10 S0873
N045 X 00.1145 F011.52 S0768
N046 X 00.0855 Z-00.2348 F010.50 S0700
N047 Z 01.4298 F200.00
N048 Z 00.1 F010.00
N049 X-00.2125 F200.00 S1024
N050 X-00.1 F010.00
N051 Z-00.0155 F015.36
N052 X 00.0575 Z-00.0575 F014.33 S0955

Figure 10.25 (Continued)

Figure 10.25 (Continued)

APT PROGRAM

As a comparison of the many computer languages besides Compact II, APT (Automatically Programmed Tools) was selected. APT was selected because it is found in many similar forms such as ADAPT, UNIAPT, and AUTOSPOT. APT has played a large role in the computer-assisted programming language market. Many other languages and CAD/CAM systems grew out of this language. This language was developed and is still heavily used in the aerospace industry due to its ability to handle multiaxis contour programming to the fifth and seventh axis level. Other languages like Compact II were only developed to do contouring moves to three axes deep, after which additional axes became positional only. It is because of this that the author feels many CAD/CAM systems have patterned themselves after the APT language. A comparison of actual computer statements generated on a CAD/CAM system will easily show this. As the APT program is reviewed, refer to the Compact II program to see the minor differences in program statements and structures.

The part to be reviewed is a simple milled part in which the periphery and a pass around the top of the part will be machined (see Figure 10.26). As in the Compact II program the part print must be reviewed, a machine selected, and setup determined before programming can start.

The machine zero is indicated on Figure 10.26 to be 2.0 in. to the left of the finish part profile on the X axis. The Y axis zero is 2.0 in. below the profile and Z zero is 1.0 in. beneath the part. To keep the process simple, it has been decided to machine one path around the periphery at full depth, and then to

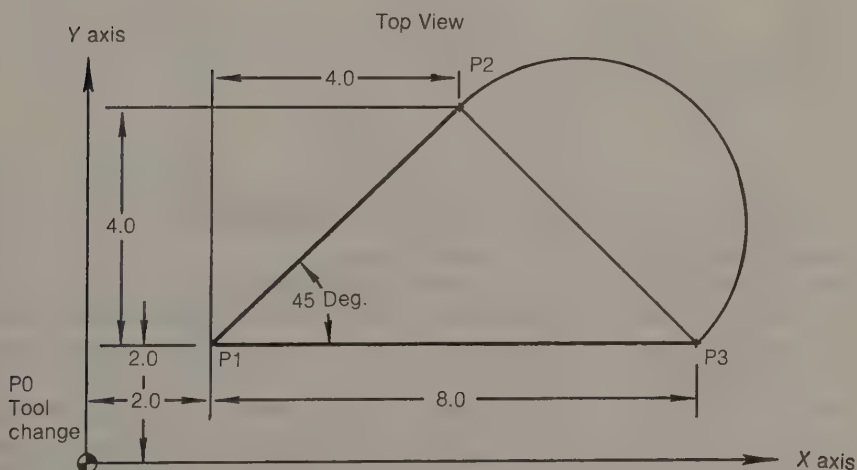


Figure 10.26 APT mill program example.

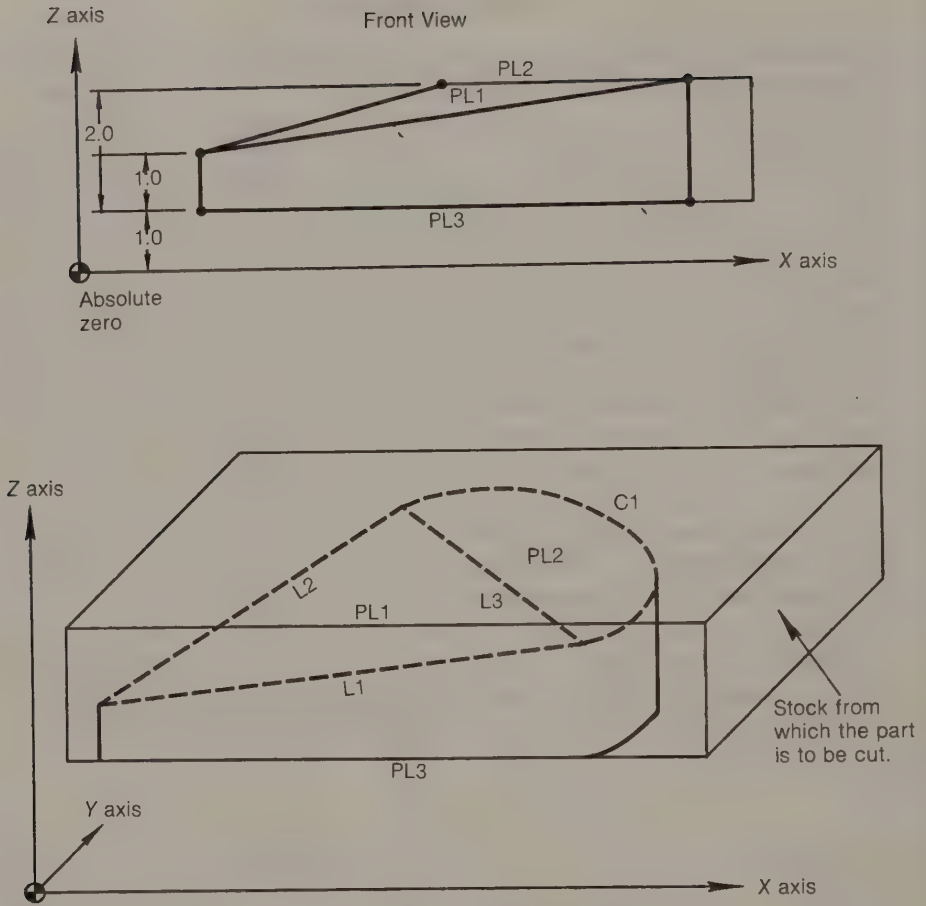


Figure 10.26 (Continued)

take one pass around the top surface. The programmer has selected a 1.0 in. diameter cutter with a 0.250 in. corner radius.

The next step in the process is to mark the geometry on the part print that needs to be defined by the programmer. Figure 10.26 shows the indicated points "P," lines "L," circles "C," and planes "PL" necessary to describe the piece part.

Once the needed geometry is determined, programming can begin. Figure 10.27 shows one programmer's solution to this particular problem. For easy reference, the program has been marked up to show its major sections. The following pages give a line by line description of the various commands in the UNIAPT program.

2466 ↑ LU, OPT=4			
INITIALIZATION Statements	{ 001 PARTNO UNAPT SAMPLE PART PROGRAM 002 \$\$ GEOMETRY 003 CLPRNT } 004 PRINT/ON }	Identification statement* Program statement Printer statements	
GEOMETRY Statements	{ 005 P0=POINT/0,0 006 P1=POINT/2,2,2 007 P3=POINT/10,2,3 010 L1=LINE/P1,P3 011 L2=LINE/P1,ATANGL,45 012 L3=LINE/(P2=POINT/6,6,3),P3 } 013 PL1=PLANE/P1,P2,P3 \$\$ CANTED PLANE ON TOP 014 PL2=PLANE/0,0,1,3 \$\$ PLANE Z=3 015 PL3=PLANE/PARLEL,PL2,ZSMALL,2 \$\$ Z=1 016 R=(LNTH(VECTOR/P2,P3))/2 017 C1=CIRCLE/TANTO,L2,XSMALL,P3,RADIUS,R 020 \$\$ 021 \$\$ MAKE PASS AROUND PERIPHERY 022 \$\$	Point definition Line definition Plane definitions Circle definition Program statement	
TOOL CHANGE Statements	{ 023 CUTTER /1,.25 \$\$DIA.=1,CORNER RADIUS=25 024 FROM /P0 025 DNTCUT }	Cutter definition	

Figure 10.27 Source program listing—UNAPT Milling Example. (*Note: 001 becomes a MACHIN/statement when programming for a specific machine tool.)

TOOL MOTION Statements	026	GOTO	/-.75,.75,1 \$\$ MOVE CLEAR OF PART	Tool motion for periphery cut
	027	GO	/L2,PL3,ON,L1,10 \$\$10 IPM FEEDRATE	
	030	CUT		
	031	TLLFT,GOLFT	/L2	
	032	GOFWD	/C1,PAST,L1	
	033	GORGT	/L1,PAST,L2	
	034	GOTO	/P0	
	035 \$\$			
	036 \$\$			
	037 \$\$			
TOOL MOTION Statements	040	GODLTA	/0,0,2	Program statement
	041	GO	/ON,L2,PL1,L1,20 \$\$ FEEDRATE=20 IPM	
	042	TLOX,GOFWD	/L2,PAST,PL2	
	043	PSIS	/PL2 \$\$ AUTOPS WOULD BE THE SAME	
	044	GOFWD	/L2,TANTO,C1	
	045	GOFWD	/C1,ON,L1	
	046	GO	/ON,L1,PL1,ON,L3 \$\$GET BACK ON PL1	
	047	INDIRV	/-1,0,0 \$\$ESTABLISH DIRECTION	
	050	GOFWD	/L1,PAST,L2	
	051	GOTO	/P0	
TERMINATION Statements	052	STOP		Tool motion for top edge
	053	END		
	054	FINI		

Figure 10.27 (Continued)

001 PARTNO UNIAPT SAMPLE PROGRAM PART

Identifies the part and serves as a title for the program listing as well as identification for the CNC control tape.

002 \$\$ GEOMETRY

The double dollar sign indicates a comment line.

003 CLPRNT

Specifies that the cutter location input data to the postprocessor is to be printed. This statement caused the X, Y, Z outputs in the following cutter location data file "CL Data" from program line numbers 24 through 51.

004 PRINT/ON

Causes surface data to be printed in its canonical form. This statement caused the output from program line numbers 005 through 17 in the CL Data file.

005 P0 = POINT/0,0

Assigns the name P0 to the point whose coordinates are $x = 0$, $y = 0$. The processor assumes $z = 0$ if only two points are given and no ZSURF statement has been given.

006 P1 = POINT/2,2,2

Assigns the name P1 to the point whose coordinates are $x = 2$, $y = 2$, $z = 2$.

007 P3 = POINT/10,2,2

Assigns the name P3 to the point whose coordinates are $x = 10$, $y = 2$, $z = 2$.

***010 L1 = LINE/P1,P3**

Defines a line through the two points P1 and P3.

011 L2 = LINE/P1, ATANGL, 45

Defines a line through the point P1 and at an angle of 45° to the X axis.

012 L3 = LINE/(P2 = POINT/6,6,3), P3

Define a line through two points P2 and P3. P2 is an example of a nested definition, i.e., it defines P2 in terms of its coordinates ($x = 6$, $y = 6$,

*Although called a line these are actually surfaces with an infinite plane perpendicular to the X-Y plane of the paper. They are called lines for convenience. It is always understood that a line in UNIAPT is really a plane. The same is true for a circle. It is actually an infinite cylinder whose sides are perpendicular to the XY, YZ, or XZ plane.

$z = 3$) within the line statement. P2 could have been written as a separate statement in which case the preceding statement would have appeared simply as $L3 = \text{LINE}/P2, P3$.

013 $\text{PL1} = \text{PLANE}/P1,P2,P3 \text{ \textit{\$ \$ CANTED PLANE ON TOP}}$

Defines a canted plane (which is the top of the part) in terms of three points: P1,P2,P3. The words following the double dollar sign are not processed by the program. They are for programmer documentation only.

014 $\text{PL2} = \text{PLANE}/0,0,1,3 \text{ \textit{\$ \$ PLANE, Z} = 3}$

Defines a plane in terms of the coefficients of the plane equation $ax + by + cz - d = 0$, where $a = 0, b = 0, c = 1, d = 3$.

015 $\text{PL3} = \text{PLANE}/\text{PARLEL},\text{PL2},\text{ZSMALL},2 \text{ \textit{\$ \$ Z} = 1}$

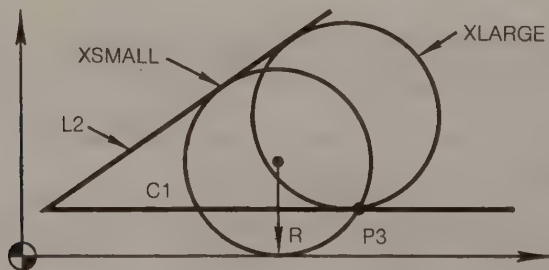
Defines a plane parallel (PARLEL) to plane (PL2) 2 units away from PL2. Two planes can meet this definition—one above and one below PL2. The modifier ZSMALL removes the ambiguity by specifying the plane with the smaller Z value: below PL2.

016 $R = (\text{LNTHF}(\text{VECTOR}/P2,P3))/2$

Defines the radius (R) by computing the length (LNTHF) of the vector (VECTOR) which connects points P2 and P3 and taking one-half of the resulting value.

017 $\text{C1} = \text{CIRCLE}/\text{TANTO},\text{L2},\text{XSMALL},\text{P3},\text{RADIUS},R$

Defines the circle (C1) in terms of line (L2) that it (C1) is tangent to and the point P3 that it passes through with a radius R. As shown in the following schematic, two circles can meet these requirements. The modifier XSMALL indicates to the system to use the one with the smallest X value.



020 $\text{\textit{\$ \$}}$

Causes a line to be skipped.

021 \$\$ MAKE A PASS AROUND PERIPHERY

A comment.

022 \$\$

Causes a line to be skipped.

023 CUTTER/1,.25 \$\$ DIA. = 1, CORNER RADIUS = .25

Defines cutter diameter (1 in.) and corner radius (0.25 in.).

024 FROM/P0

Defines cutting tool beginning position as being at point (P0). To see tool path generated by computer statement refer to Figure 10.28.

025 DNTCUT

Indicates that tool center information for motion statements is not to be passed on to the postprocessor.

026 GOTO/-.75,.75,1 \$\$ MOVE CLEAR OF PART

Move cutter (clear of part) tool to $x = -0.75$, $y = 0.75$, and $z = 1$.

027 GO/L2,PL3,ON,L1,10 \$\$ 10 IPM FEEDRATE

Move cutting tool to the position where L2, PL3, and L1 intersect. PL3 is established as the part surface (the second surface in the command) for all subsequent motion commands. This command also sets the feed-rate to 10 ipm.

030 CUT

Indicates that tool center information for motion statements is to be passed on to the postprocessor. It causes cutter location data output to begin with preceding statement.

031 TLLFT, GOLFT/L2

Position the tool to left (TLLFT) of L2 and move it along the intersection of surfaces L2 and PL3 to C1. A check surface is implied, which means to use the drive surface from the next command for the check surface for this command. Also a TANTO condition is indirectly specified since the next command is a GOFWD (*drive surface*: surface which tool is moved along; *check surface*: surface which tool will stop at.)

032 GOFWD/C1,PAST,L1

Move tool along intersection of surfaces C1 and PL3 until past the plane L1.

033 GORGT/L1,PAST,L2

Turn right and move tool along intersection of L1 and PL3 past the plane L2.

034 GOTO/P0

Move tool from its present position to P0 ($x = 0$, $y = 0$, $z = 0$).

035 \$\$

A remark—causes a line space.

036 \$\$ MAKE A PASS AROUND TOP OF PART

A remark. Notes that the next group of statements will begin machining the part surface top.

037 \$\$

A line space.

040 GODLTA/0,0,2

Adds the indicated increments to the tool position. Since x and y are zero, tool is moved straight up 2 in. To see tool path generated by computer statement, refer to Figure 10.29.

041 GO/ON,L2,PL1,L1,20 \$\$ FEEDRATE = 20 IPM

Position the tool at the intersection of drive surface L2, part surface PL1, and check surface L1. Tool end is centered on the L2 surface and to the PL1 and L1 surfaces. Feedrate is set at 20 ipm. PL1, since it is the second surface in the GO/ command, is established as the modal part surface.

042 TLON, GOFWD/L2,PL2

Tool is centered on (TLON) the drive surface which is L2 in this statement example. Go forward (GOFWD) along the intersection of L2 and PL1, which is the part surface. Stop at PL2, which is the check surface.

043 PSIS/PL2 \$\$ AUTOPS WOULD BE THE SAME

Causes part surface to be changed to PL2. PSIS means "part surface is." The comment refers to the command AUTOPS, which can be used to establish a plane at the cutters current z height.

044 GOFWD/L2,TANTO,C1

Move tool along the intersection of L2 and PL2 until tangent to circle C1.

045 GOFWD/C1,ON,L1

Move tool along the intersection of C1 and PL2. Stop on surface L1.

046 GO/ON,L1,PL1,ON,L3 \$\$ GET BACK ON PL1

Move tool to the position where L1, PL1, and L3 intersect. The "ON" modifier means the tool center is to be positioned "on" L1 and "on" L3.

047 INDIRV/-1,0,0 \$\$ ESTABLISH DIRECTION

This command, in the direction of vector, establishes a tool direction as being in the minus x direction ($x = -1$, $y = 0$, $z = 0$).

050 GOFWD/L1,PAST,L2

Go forward along the intersection of L1 and PL1 and stop just past the surface L2.

051 GOTO/P0

A point to point command. Position tool from where it is to point P0.

052 STOP

A postprocessor command that stops the machine tool and machine tool controller input reader. Feedrate is reduced to zero and spindle and coolant are turned off.

053 END

A postprocessor command that signals the end of a logical section of a part program.

054 FINI

The end of the complete part program.

In an APT program's processing, after the programmer creates the computer source program, it is submitted to the computer. The computer upon receipt of source information checks to see if it is a valid statement. Once validity is established, the computer creates the cutter location file data either by interaction or through batch processing. The CL Data file consists of all the geometry and cutter location information required to machine the part in coordinate numerical form. This information can then be transformed into the correct numerical control format for the particular machine tool and CNC control unit. The information in Figure 10.30 is a printout of the CL Data from the UNIAPT sample program. Note: For the readers benefit, the axis designators are shown in parentheses.

After computer processing has occurred to the point of the CL Data file just reviewed, the programmer can then request tool path graphics from the system.

Source Program Line #

```

REC0005 SURFACE=POINT P0
  (X) .000000 (Y) .000000 (Z) .000000
REC0006 SURFACE=POINT P1
  (X) 2.000000 (Y) 2.000000 (Z) 2.000000
REC0007 SURFACE=POINT P3
  (X) 10.000000 (Y) 2.000000 (Z) 3.000000
REC0010 SURFACE=LINE L1
  (X) .000000 (Y) 1.000000 (X) .000000 (Y) 2.000000
REC0011 SURFACE=LINE L2
  (X) .707106 (Y) -.707106 (X) .000000 (Y) .000000
REC0012 SURFACE=POINT P2
  (X) 6.000000 (Y) 6.000000 (Z) 3.000000
REC0012 SURFACE=LINE L3
  (X) .707106 (Y) .707106 (X) .000000 (Y) 8.485282
REC0013 SURFACE=PLANE PL1
  (X) -.123091 (Y) -.123091 (Z) .984731 1.477097
(X, Y, Z values of unit vector perpendicular to plane, fourth value is
minimum distance of plane from origin.)
REC0014 SURFACE=PLANE PL2
  (X) .000000 (Y) .000000 (Z) 1.000000 (Z) 3.000000
(First three values establish plane vector.)
REC0015 SURFACE=PLANE PL3
  (X) .000000 (Y) .000000 (Z) 1.000000 (Z) 1.000000
REC0016 SURFACE=VECTOR
  4.000000 -4.000000 .000000
REC0016 SURFACE=SCALAR R
  2.828427
REC0017 SURFACE=CIRCLE C1
  (X) 8.000000 (Y) 3.999999 (Z) .000000 (X1) .000000
  (Y1) .000000 (Z1) 1.000000 (Radius) 2.828427
(X1, Y1, Z1 form unit vector of cylinder passing through circle.)
REC0024
  (X) .000000 (Y) .000000 (Z) .000000
REC0030
  (X) 1.292893 (Y) 2.000000 (Z) 1.000000
REC0031
  (X) 5.646446 (Y) 6.353553 (Z) 1.000000
REC0032
  (X) 5.687013 (Y) 6.394128 (Z) 1.000000
  5.770902 6.472423 1.000000
  5.857440 6.547782 1.000000

```

Figure 10.30 Program output; surface data in canonical form and cutter location data. (Note: Axis call labels will not be printed on computer generated readouts.)

5.946523	6.620113	1.000000
6.038047	6.689330	1.000000
6.131902	6.755352	1.000000
6.227977	6.818100	1.000000
6.326156	6.877500	1.000000
6.426326	6.933480	1.000000
6.528365	6.985975	1.000000
6.632152	7.034921	1.000000
6.737565	7.080262	1.000000
6.844478	7.121942	1.000000
6.952764	7.159913	1.000000
7.062294	7.194129	1.000000
7.172939	7.224550	1.000000
7.284566	7.251140	1.000000
7.397043	7.273866	1.000000
7.510238	7.292701	1.000000
7.624013	7.307626	1.000000
7.738235	7.318619	1.000000
7.852769	7.325669	1.000000
7.967477	7.328768	1.000000
8.082225	7.327911	1.000000
8.196873	7.323100	1.000000
8.311290	7.314340	1.000000
8.425335	7.301643	1.000000
8.538875	7.285022	1.000000
8.651775	7.264497	1.000000
8.763900	7.240094	1.000000
8.875118	7.211841	1.000000
8.985297	7.179771	1.000000
9.094303	7.143923	1.000000
9.202010	7.104339	1.000000
9.308289	7.061067	1.000000
9.413013	7.014158	1.000000
9.516059	6.963666	1.000000
9.617302	6.909654	1.000000
9.716624	6.852184	1.000000
9.813907	6.791326	1.000000
9.909034	6.727149	1.000000
10.001891	6.659734	1.000000
10.092372	6.589157	1.000000
10.180365	6.515504	1.000000
10.265769	6.438862	1.000000
10.348479	6.359323	1.000000

Note: Amount of
output will
depend on path
tolerance setup in
computer.

Figure 10.30 (Continued)

10.428398	6.276979	1.000000
10.505432	6.191930	1.000000
10.579490	6.104276	1.000000
10.650482	6.014123	1.000000
10.718326	5.921575	1.000000
10.782939	5.826745	1.000000
10.844246	5.729745	1.000000
10.902172	5.630689	1.000000
10.956650	5.529695	1.000000
11.007617	5.426883	1.000000
11.055007	5.322376	1.000000
11.098769	5.216298	1.000000
11.138848	5.108775	1.000000
11.175198	4.999934	1.000000
11.207775	4.889905	1.000000
11.236541	4.778819	1.000000
11.261459	4.666807	1.000000
11.282504	4.554002	1.000000
11.299648	4.440540	1.000000
11.312870	4.326555	1.000000
11.322158	4.212180	1.000000
11.327496	4.097554	1.000000
11.328883	3.982812	1.000000
11.326312	3.868091	1.000000
11.319790	3.753526	1.000000
11.309322	3.639254	1.000000
11.294922	3.525411	1.000000
11.276608	3.412132	1.000000
11.254401	3.299551	1.000000
11.228325	3.187802	1.000000
11.198414	3.077018	1.000000
11.164703	2.967332	1.000000
11.127232	2.858872	1.000000
11.086044	2.751768	1.000000
11.041190	2.646148	1.000000
10.992722	2.542136	1.000000
10.940697	2.439856	1.000000
10.885180	2.339430	1.000000
10.826233	2.240978	1.000000
10.763928	2.144615	1.000000
10.698340	2.050457	1.000000
10.629544	1.958615	1.000000
10.557624	1.869199	1.000000

Figure 10.30 (Continued)

10.482666	1.782316	1.000000
10.404758	1.698067	1.000000
10.323992	1.616553	1.000000
10.240463	1.537871	1.000000
10.197368	1.500000	1.000000
REC0033		
(X) .792893	(Y) 1.500000	(Z) 1.000000
REC0034		
(X) .000000	(Y) .000000	(Z) .000000
REC0040		
(X) .000000	(Y) .000000	(Z) 2.000000
REC0041		
(X) 1.500000	(Y) 1.500000	(Z) 1.923070
REC0042		
(X) 5.807718	(Y) 5.807718	(Z) 2.999999
REC0044		
(X) 6.000000	(Y) 5.999999	(Z) 2.999999
REC0045		
(X) 6.037401	(Y) 6.037408	(Z) 2.999999
6.114954	6.109367	2.999999
6.195144	6.178376	2.999999
6.277858	6.244339	2.999999
6.362981	6.307162	2.999999
6.450393	6.366759	2.999999
6.539973	6.423045	2.999999
6.631595	6.475944	2.999999
6.725130	6.525378	2.999999
6.820449	6.571281	2.999999
6.917417	6.613588	2.999999
7.015899	6.652239	2.999999
7.115758	6.687181	2.999999
7.216853	6.718365	2.999999
7.319044	6.745746	2.999999
7.422187	6.769288	2.999999
7.526138	6.788957	2.999999
7.630752	6.804725	2.999999
7.735883	6.816570	2.999999
7.841382	6.824476	2.999999
7.947103	6.828432	2.999999
8.052899	6.828432	2.999999
8.158621	6.824476	2.999999
8.264121	6.816570	2.999999
8.369251	6.804725	2.999999

Figure 10.30 (Continued)

8.473865	6.788957	2.999999
8.577816	6.769288	2.999999
8.680958	6.745746	2.999999
8.783149	6.718364	2.999999
8.884245	6.687180	2.999999
8.984103	6.652238	2.999999
9.082585	6.613586	2.999999
9.179553	6.571279	2.999999
9.274872	6.525377	2.999999
9.368407	6.475942	2.999999
9.460029	6.423044	2.999999
9.549609	6.366757	2.999999
9.637020	6.307160	2.999999
9.722144	6.244336	2.999999
9.804858	6.178374	2.999999
9.885048	6.109365	2.999999
9.962601	6.037406	2.999999
10.037410	5.962597	2.999999
10.109370	5.885043	2.999999
10.178378	5.804853	2.999999
10.244339	5.722139	2.999999
10.307163	5.637016	2.999999
10.366760	5.549604	2.999999
10.423046	5.460023	2.999999
10.475944	5.368402	2.999999
10.525379	5.274866	2.999999
10.571282	5.179548	2.999999
10.613589	5.082580	2.999999
10.652240	4.984098	2.999999
10.687182	4.884239	2.999999
10.718365	4.783143	2.999999
10.745747	4.680953	2.999999
10.769290	4.577810	2.999999
10.788958	4.473859	2.999999
10.804725	4.369245	2.999999
10.816570	4.264114	2.999999
10.824477	4.158614	2.999999
10.828433	4.052892	2.999999
10.828433	3.947097	2.999999
10.824477	3.841376	2.999999
10.816570	3.735876	2.999999
10.804725	3.630746	2.999999
10.788956	3.526132	2.999999

Figure 10.30 (Continued)

10.769287	3.422181	2.999999
10.745746	3.319038	2.999999
10.718364	3.216847	2.999999
10.687179	3.115752	2.999999
10.652237	3.015893	2.999999
10.613586	2.917411	2.999999
10.571278	2.820443	2.999999
10.525375	2.725125	2.999999
10.475940	2.631589	2.999999
10.423042	2.539968	2.999999
10.366756	2.450388	2.999999
10.307159	2.362976	2.999999
10.244335	2.277853	2.999999
10.178372	2.195139	2.999999
10.109363	2.114950	2.999999
10.037403	2.037396	2.999999
10.000000	2.000000	2.999999
REC0046		
(X)10.000000	(Y)2.000000	(Z)3.048070
REC0050		
(X)1.292893	(Y)2.000000	(Z)1.959681
REC0051		
(X).000000	(Y).000000	(Z).000000
OPT=		

Figure 10.30 (Continued)

Graphics such as those in Figures 10.28 and 10.29 will either be drawn on a plotter or displayed on a CRT screen depending on the system. Note though that the computer system does not normally indicate cross reference numbers of the program statements.

The next computer processing step will be for the CL Data to be postprocessed into a machine control program file. During postprocessing the computer converts the CL Data into the proper format to be understood by the machine tool control unit. At this time the postprocessor also adds additional function codes to specify types of motion, speeds, and feeds and auxiliary codes for turning program options on and off. Figure 10.31 shows a typical machine tool output or tape file generated by the postprocessor for our UNIAPT sample program.

GRAPHICS-BASED SYSTEMS

Graphics-based systems are referred to as Graphical Numerical Control (GNC). They differ from language-based systems in the following manner.

N010 G00 X0 Y0 Z10 S400 M03	Check home position P0 start spindle
N020 X-.75 Y.75 Z1 M08	Move clear of part, set Z depth
N030 G01 X1.2929 Y2.000 F10.0	Feed to Ref. 27, Figure 10.10
N040 X5.6464 Y6.3536	Feed to Ref. 31
N050 G02 X8.0 Y7.3284 I2.3536 J.9748	Cut circle 1
N060 X11.3284 Y4.00 I0 J3.3284	
N070 X10.1974 Y1.5 I3.3284 J0	Cut circle to Ref. 32
N080 G01 X.7929	Feed to Ref. 33
N090 G00 Z3.1 M09	Retract Z, turn coolant off
N100 X0 Y0 Z10	Rapid to P0 Ref. 34
N110 Z2.0	Rapid position Z Ref. 40
N120 G01 X1.5 Y1.5 Z1.9231 F20 M08	Feed to Ref. 41 turn coolant on
N130 X5.8077 Y5.8077 Z3.0	Feed to Ref. 42
N140 X6.0 Y6.0	Feed to Ref. 44
N150 G02 X8.00 Y6.8284 I2.0 J.8284	Cut circle 1 top
N160 X10.8284 Y4.0 I0 J2.8284	Cut circle 1
N170 X10.0 Y2.0 I2.8284 J0	Cut circle 1 Ref. 45
N180 G01 Z3.0481	Position Z axis Ref. 46
N190 X1.2929 Z1.9597	Feed to position Ref. 50
N200 G00 Z3.1 M09	Retract Z position Ref. 50, turn coolant off
N210 X0 Y0 Z10 M05	Rapid to home position P0, turn spindle off
N220 M30	End of program

Figure 10.31 Postprocessor output UNIAPT sample program.

When the geometric detail of the component has been constructed on the CRT screen, the outline shape of the cutting tool, or tools, to be used are superimposed on the component image and can be freely moved around using the cursor control device. Thus the programmer can, in effect, select appropriate tool paths to facilitate machining of the component or component detail. A better impression of the process may be obtained by considering the following example.

Figure 10.32 illustrates a component detail as it might be “drawn” on the computer screen. Also shown are two different-sized circles representing the superimposed cutters required to machine the outside profile and clear the inner shape, which represents a pocket.

Consider the large-diameter cutter first. This cutter is to be used to machine the outside profile. The size and type of cutter will have been established already and entered into the data file, possibly via a keyboard entry. The programmer now has free control to move the cutter to any position he or she chooses.

The cutter is first positioned in a suitable starting position. If cutter radius

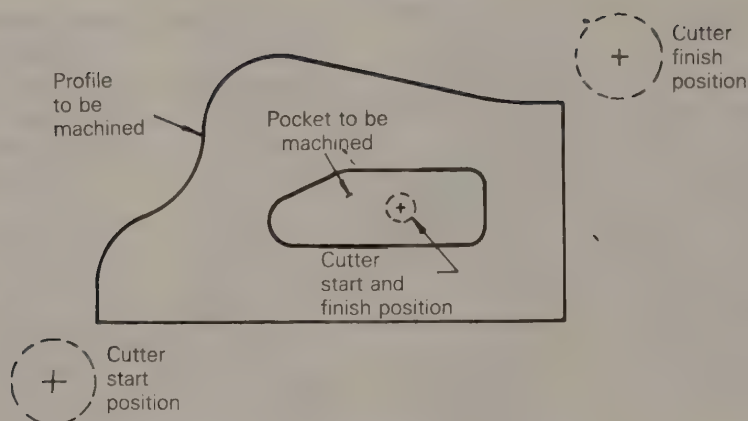


Figure 10.32 *A graphical numerical control display.*

compensation is required, then the starting position will be a suitable distance away from the profile so that the cutter can attain the correct position before contacting the work surface. This start position is then entered into the data file either via the cursor control or the keyboard.

Having established the start position the programmer moves the cutter image to the finish position, and this too is entered into the file.

Using this very simple data entry the computer is able to compute the complete cutter route, indicating it on the screen alongside the component. It will also list, in numerical form, the cutter location in relation to a previously established datum at each stage in the machining process; the list appears on a second CRT screen or an interfaced printer.

Now consider the machining of the pocket. To achieve this the programmer simply has to move the cutter to the start position at the center of the pocket as indicated. All that is now needed are keyboard data entries to establish that a pocket clearance routine is required. From this data entry the computer will establish all the moves necessary to clear the pocket and machine the final profile. Again, the cutter paths will be shown graphically and also presented in numerical form, the dimensional values being in relation to a predetermined datum.

Note that in this simple description no mention has been made of movement in the third, that is, Z axis. Clearance in the Z axis, and the depth to which the cutter is required to go, will be entered via the keyboard as the program is constructed. Data relating to speeds and feeds will be entered in the same way.

In addition to the component image displayed on the screen, it is possible to include any workholding or tool-holding arrangements that may influence the choice of tool paths. An example of such a display for a turning operation is shown in Figure 10.33. By including such detail it is possible for the pro-



Figure 10.33 Part programming station showing use of computer graphics for program proving.

grammer to be reasonably confident that all programmed moves will be collision-free.

As with language-based systems, it will be possible to obtain copies of the data entered via an interfaced printer.

Data can also be reprocessed to verify its accuracy prior to postprocessing into machine control language. Graphics-based systems have also given way to the more powerful CAD/CAM systems that can generate commands through graphics for all axes.

CAD/CAM: COMPUTER-AIDED DESIGN AND MANUFACTURE

The expression “computer-aided design,” or more commonly the abbreviation CAD, is the term used to describe the process by which engineering designers use the computer as a creative tool allowing them to produce, evaluate, modify, and finalize their designs. The computer becomes a terminal at which the de-

signer sits to analyze data, make calculations, and use the computer graphics to build up quickly and efficiently a three-dimensional image of a projected design. The image can be rotated and viewed from different angles, sectioned through various planes, stretched, condensed, and generally assessed. Modifications can be made instantly. As each stage of the design process proceeds, the resulting data can be stored and retrieved at will.

When the designer is satisfied, the details of the design can be transferred electronically to the drafting department. Here the draftsman or draftswoman transforms the original designs into a series of engineering drawings which he or she creates on the computer screen, and again at this stage each individual component can be rotated, sectioned, scaled up or down and so on in a further process of evaluation which may or may not result in modifications to the original design. When this task is finished, fully dimensioned drawings can be printed from an interfaced printer or plotter, or alternatively the information can be stored as numerical data for later retrieval. The use of computers as an aid to manufacturing processes is referred to as Computer-Aided Manufacturing (CAM). The CAPP process is part of that general definition. Together CAD and CAM form the basis of Computer-Aided Engineering (CAE).

From the foregoing text the reader will now appreciate that an important element of the CAPP process is the geometric definition of the component detail. It is, of course, also central to any computer-aided drafting process. It is logical therefore that the two processes should be linked. Most CAD/CAM software currently in use provides this facility.

The transfer involves reducing the detail drawing (Figure 10.39) to its basic geometry by removing all dimensions, notes, leader lines etc. This is easily and instantly achieved, since the drawing will have been compiled by the use of overlays or layers, with one layer containing the basic drawing and subsequent layers containing dimensions and other data. Each layer is capable of being displayed independently of the others.

With the drawing reduced to its basic form, the geometry created as part of the drafting process can be used for part programming purposes, thus eliminating the geometry construction stage of the CAPP process, and speeding part programming activity considerably.

Computer-aided manufacturing (CAM) is the term generally used to describe manufacturing processes that are computer controlled. One very important manufacturing process is metal cutting, and the computer involvement in this area of activity has been the subject of this text. Metal cutting is, however, just one type of manufacturing process that is computer controlled. There are many others: welding, flame cutting, presswork, electrodischarge machining, parts assembly, and so on.

All the processes listed previously are truly manufacturing processes, that is, the end result is a component or an assembly of components. But there are a host of other essential functions that play a part in the overall setup. The supply of materials and tooling, part programming, process control, and in-

spection/quality control, are some functions that are workshop-related. Spreading the net further, there are the financial aspects—marketing, stock control, and distribution, for example—and of course there is the design and drafting process discussed earlier. It is possible for all these functions to be interrelated via computer control into a total computerized manufacturing system referred to as CIM (Computer-Integrated Manufacturing).

Since it is the practical aspects of the system that are most likely to be of interest to the reader, that is, the design and making of the product, the relationships between these two areas are worthy of further comment and are perhaps the key elements in the system. In the past, CAD and CAM, that is the production element of CAM, have developed as two separate activities, with the application of computers to the production process being somewhat ahead of CAD. Increasingly, even in small companies, they are now being seen not as two related functions but as one integrated function. Already the more sophisticated design/drafting systems are linked to the manufacturing process via part-programming facilities. Manufacturing aspects are fully considered at the design stage, and machine-control programs are produced direct from design data rather than from a separate, and therefore error prone, analysis of an already finalized engineering drawing. The process may also, ultimately, eliminate the need for conventional drawings. Totally computerized engineering has arrived, and the rate at which it is implemented, especially in large companies, is likely to be rapid.

CAD/CAM systems like computer-assisted programming systems can be found with many different capabilities running on a multitude of platforms. Capabilities run from low-level two-dimensional to high-level three-dimensional systems. The hardware used ranges from small personal computers through minis and mainframes. All systems, though, will have a keyboard for operator commands, a user interface device (mouse, joystick, puck and tablet, or light pen), graphics screen, and central processing unit (CPU) with an information storage media device like a floppy disk or magnetic tape (Figure 10.34). For discussion purposes in this book we will be looking at a mini-based system that uses a mouse-actived command system displayed on a CRT screen. Figure 10.35 shows the area of the screen where various information will be displayed.

User Interface

CAD/CAM systems normally use some form of standard user interface, which essentially replaces direct keyboard input to an applications program with six types of interactive input:

- interactive headline
- single keyboard input
- graphic icons
- pointer strokes
- pop-up menus
- text macro processor

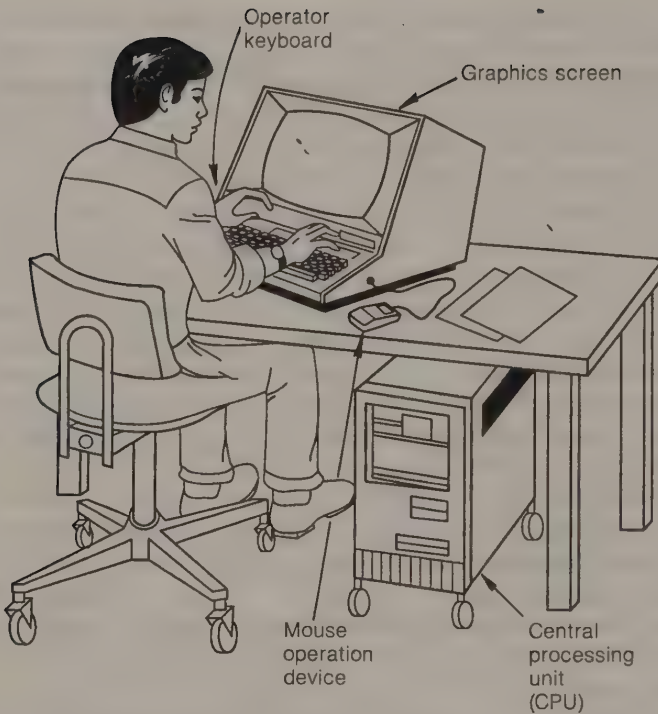


Figure 10.34 CAD/CAM (Computer-Aided Design/Computer-Aided Manufacturing) workstation.

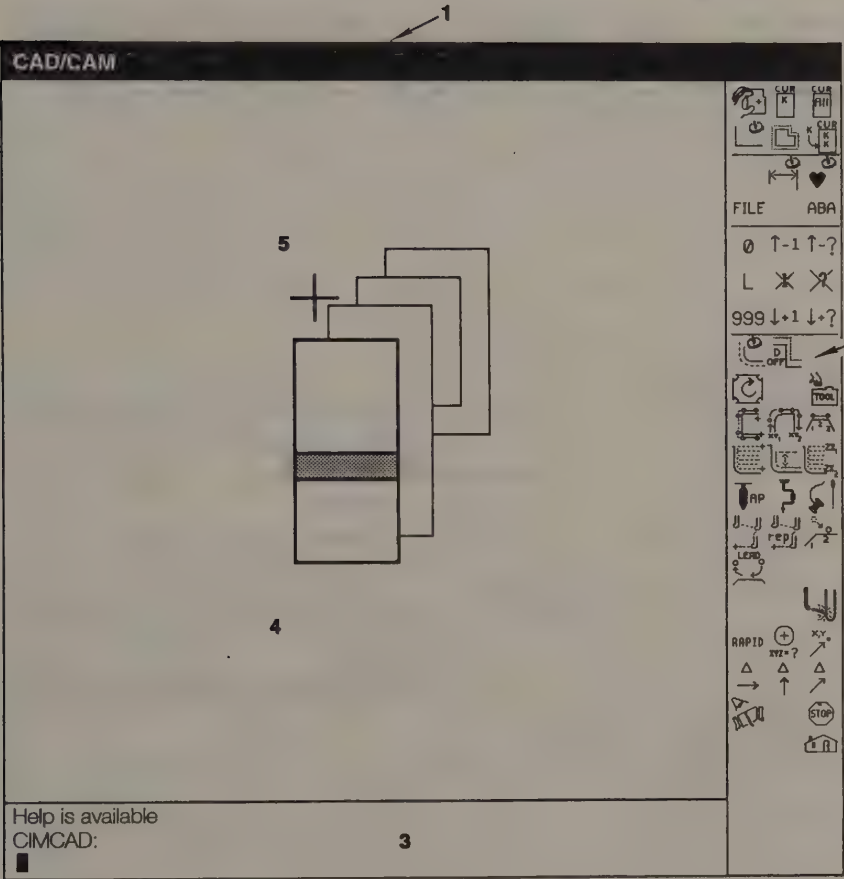
Graphic icons are always visible and reside in a fixed location at the right side of the user CRT screen and are activated through the use of the mouse instead of keying in information. Other systems may use electronic pucks, joysticks, or light pens to select commands from the CRT screen or a tablet. Pop-up menus are a method of command input by which text or symbols representing commands are selected, rather than typing in the command itself. When the *middle mouse button* (MMB) (Fig. 10.37) is pressed, pop-up menus appear on the screen near the screen pointer position. Both menus and icons are user-definable and may be modified at any time during a session. By providing both graphic icons and pop-up menus, the user's focus remains on the screen.

The mouse-controlled screen pointer is used to select items from the pop-up menus and graphic icons, and for input of coordinating information. The information is added by pressing the *left mouse button* (LMB) (Figure 10.36).

Single keystrokes by keyboard input can be used to invoke a particular effect, such as inputting a new geometric point at exactly the same location as an existing point near the screen pointer.

As in all application programs that use various forms of user interface, all menu selections and single-keystroke commands are converted into long com-

CIM CAD USER ENVIRONMENT



- 1. HEADLINE
- 2. GRAPHIC ICONS
- 3. SCROLL AREA
- 4. GRAPHICS AREA
- 5. POP-UP MENUS

Figure 10.35 CAD/CAM work station screen display. (Courtesy of CIMLINC Inc.).

puter commands for execution. It is these long commands that form the basis of the applications program and out of which any new user menus or single-keystroke commands are built.

Cursor and Mouse Concepts

A cursor is a symbol displayed on the CRT screen. Cursors vary in style depending on which mouse (or activation device) button is pressed and the menu being used. The style of the cursor has a specific meaning. It functions as a pointing device and is positioned by moving the mouse or activation device. Sometimes you use it to point to a location where you wish to start a line or specify the next position on a line. Other times you use it to make a selection from a menu. In many cases you use it to indicate which entity is used to perform a specific function, which is related to the entity near the cursor, by a command chosen from one of the many available menus. *The position of the cursor is very important in these conditions.*

Left Mouse Button (LMB)—GRAPHICS (Working) Cursor The graphics cursor (normally a plus sign) is used as a pointing device to create and modify the entities (lines, text, machining paths, etc.), which define the graphics. It has other uses also, which are addressed further on in this section. The style of the graphic cursor may be changed by the user.

Middle Mouse Button (MMB)—Pop-Up Cursor Pop Up cursors are used to make a selection from the Pop Up menus. Pressing the MMB activates and displays the various Pop Up menus, *only for as long as the MMB is held down*. They appear near the graphics cursor position at the time the MMB is pressed. When it is released, the pop up menus disappear.

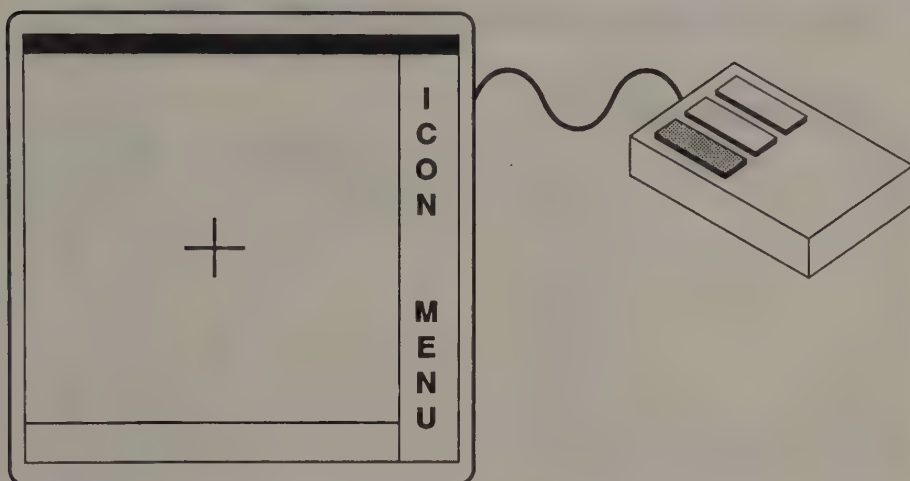
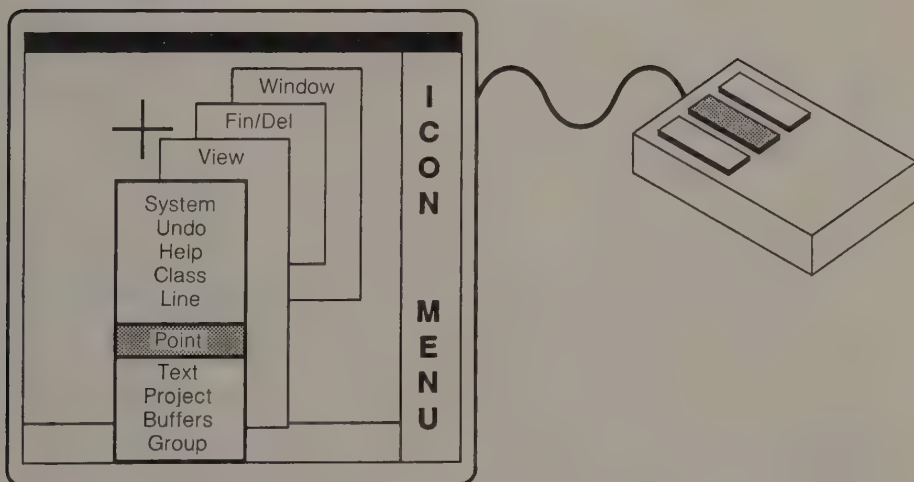
On the CIMLINC system shown the TITLES are displayed in inverse video (white letters black background).

As the mouse is moved within the menus, the selection the cursor is on turns to inverse video. When the cursor is positioned on a menu that is underneath another, it instantly pops to the top as shown in Fig. 10.37. In cases where the menu is small, position the cursor in the title area to pop it to the top, then on the desired selection of the menu. Releasing the button on a highlighted command will activate the command without typing.

Right Mouse Button (RMB)—Static Menu (ICON) Cursor The static icons are graphic representations of various computer commands, either CAD or CAM, that can be cursor-selected to keep typing to a minimum. By placing the commands on the computer screen instead of a desk tablet device, the operator's eyes do not have to leave the screen.

The icon cursor is an inverse video of a specific icon (see Figure 10.38). It is used to make selections from the static menu.

When the RMB is pressed, the cursor is displayed instantly in the static menu (icon menu), for as long as pressure is applied. When the RMB is released,

Left Mouse Button (LMB) — GRAPHICS (Working) Cursor**Figure 10.36****Middle Mouse Button (MMB) — Pop-Up cursor****Figure 10.37**

Right Mouse Button (RMB) ← Static Menu (ICON) Cursor

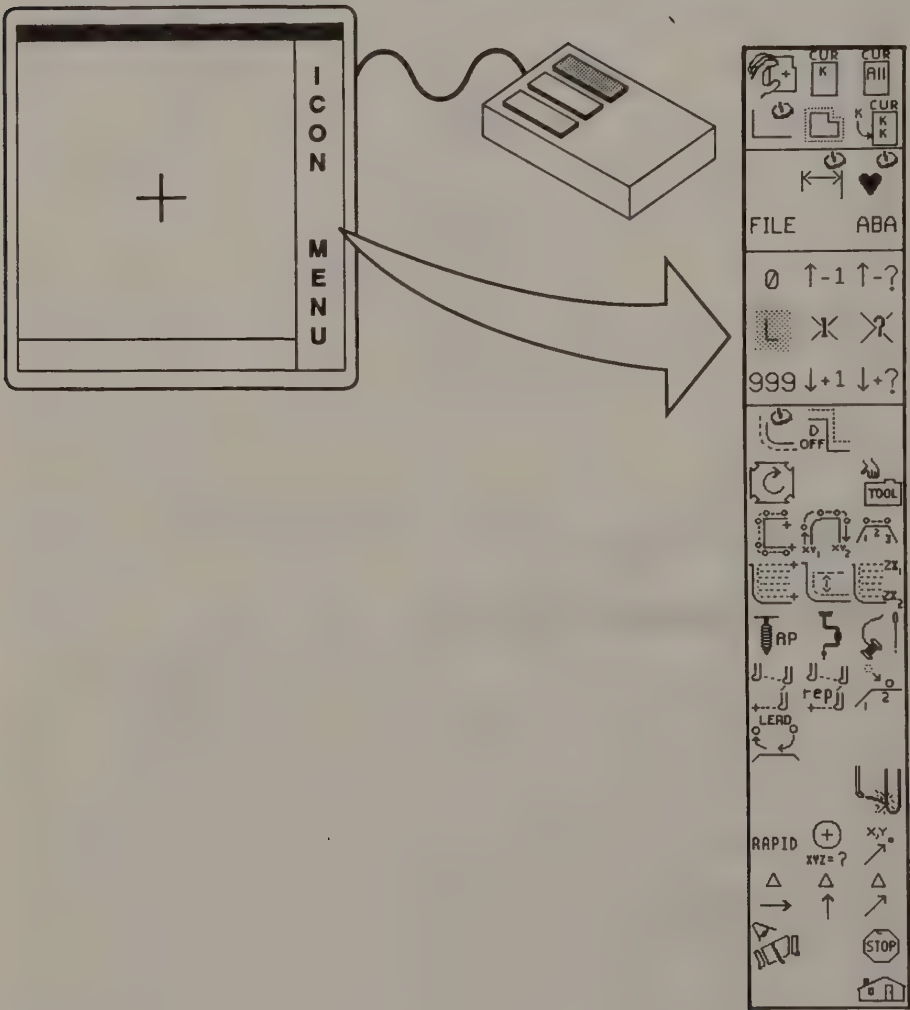


Figure 10.38

the command or action associated to the icon is issued to the CAD/CAM control processor unit and the actual printed command appears at the bottom of the screen.

Now that we have established a general definition of a CAD/CAM system, let us follow the complete process through of generating a program for a part on a CNC lathe. The first step is to generate a CAD design of the part to be manufactured. Figure 10.39 shows a screen display of the screw jack part once it is completed.

After the part creation in the CAD system is complete, the programmer must process the part for its machining operations deciding on the tools to be used. For the screw jack the following tools were decided upon.

- 80° diamond-shaped-insert right-hand turning tool for rough facing and turning
- 0.500 in.-diameter drill for hole
- 0.125-in.-width parting tool for grooves or thread relief
- 30° diamond-shaped-insert right-hand turning tool for finish facing and turning
- Brazed carbide-tipped boring bar for rough and finish boring of tapered bore
- 60° threading tool for 1-24 UNF-2B thread
- 0.125-in.-width parting tool for part cutoff.

Once processing is complete, the programmer will check the tool library generated on the computer system to see if additional tools need to be generated. Figure 10.40 illustrates the necessary geometry and origin point that need to be created in CAD so tool information can be transferred into the tool library in CAM. Having this information stored in the CAM system allows programmers to create and display a variety of tooling packages for use on various part programs.

It has been shown that tools can be drawn up in CAD and its computer definition transferred to a tool library for later use in CAM. It should be noted, though, that tools can also be defined in the CAM system through the use of special icons and pop up menus such as those that appear in Figure 10.41.

Now that the tooling is complete, the programmer will reduce the part drawing to only the machining curves that are required to produce the part program. The programmer reduces the drawing, as shown in Figure 10.42, to its basic profile and then the system will convert them to machining curves with some basic entered commands. Figure 10.42 shows the labeled machining curves that appear as a profile of half the part. Curve K1 is the internal bore, K2 is the part center, K3 is the outside profile, and K99 is the original part stock outline.

Figure 10.43 shows us that the machining curves are made up of straight and circular line segments called spans. The CAM system gives us the freedom

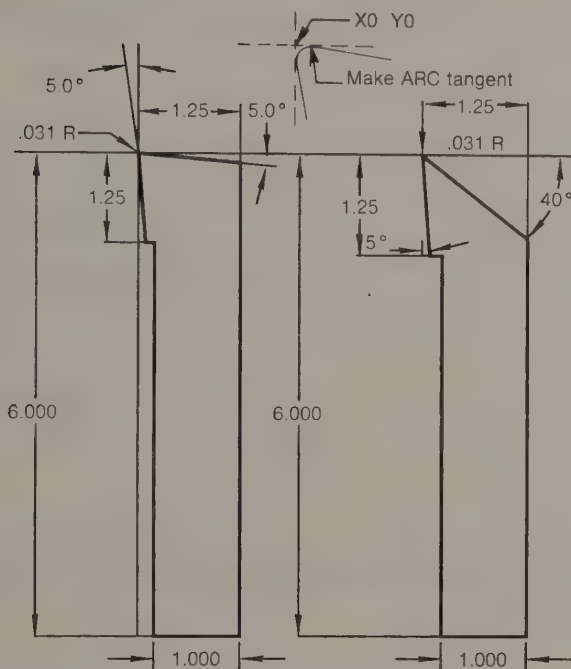
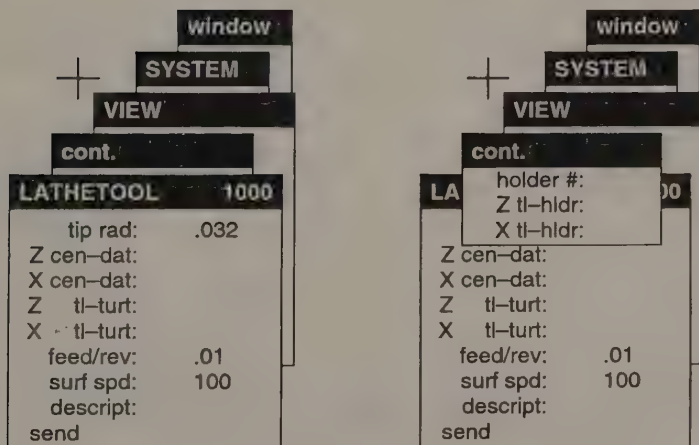


Figure 10.40 Cutting tool geometry creation for tool library.



(a)



(b)

Figure 10.41 CAM (a) tool definition icons and (b) pop up menus.

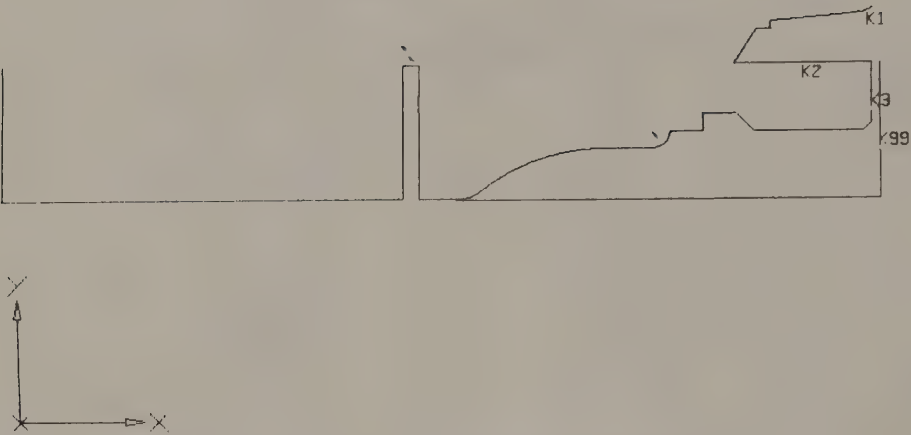


Figure 10.42 CAD part drawing reduced to required CAM profile curves.

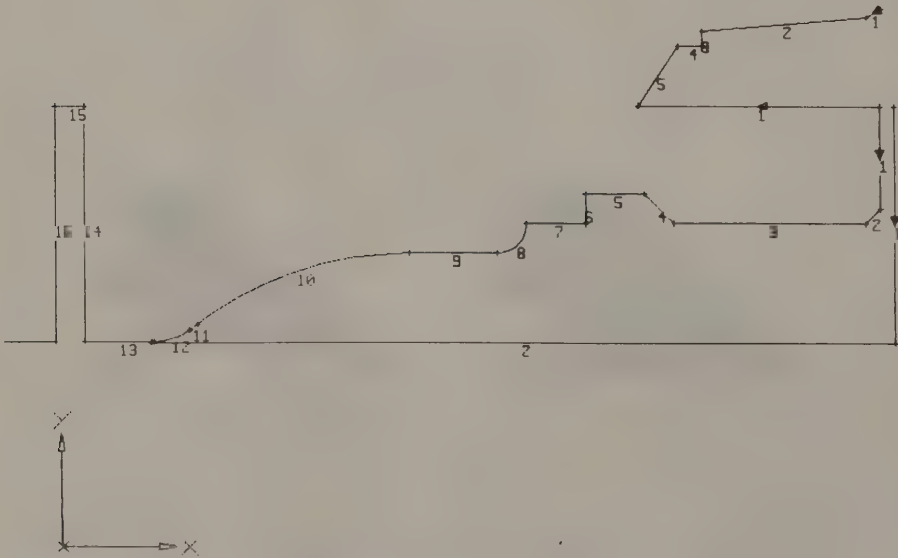


Figure 10.43 CAM profile curves displayed in spans or segments form.

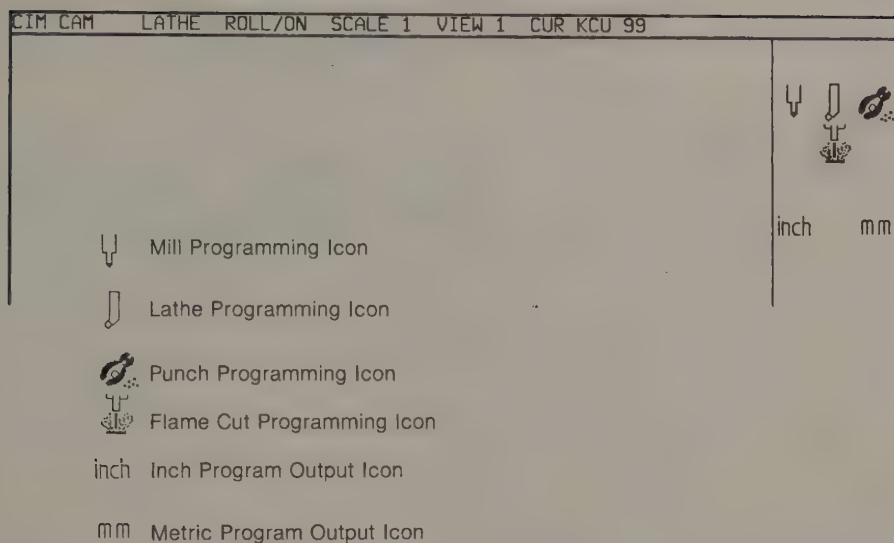


Figure 10.44 CRT display entering CAM for machining option and inch/metric output. a—mill programming icon; b—lathe programming icon; c—punch programming icon; d—flame cut programming icon; e—inch program output icon; f—metric program output icon.

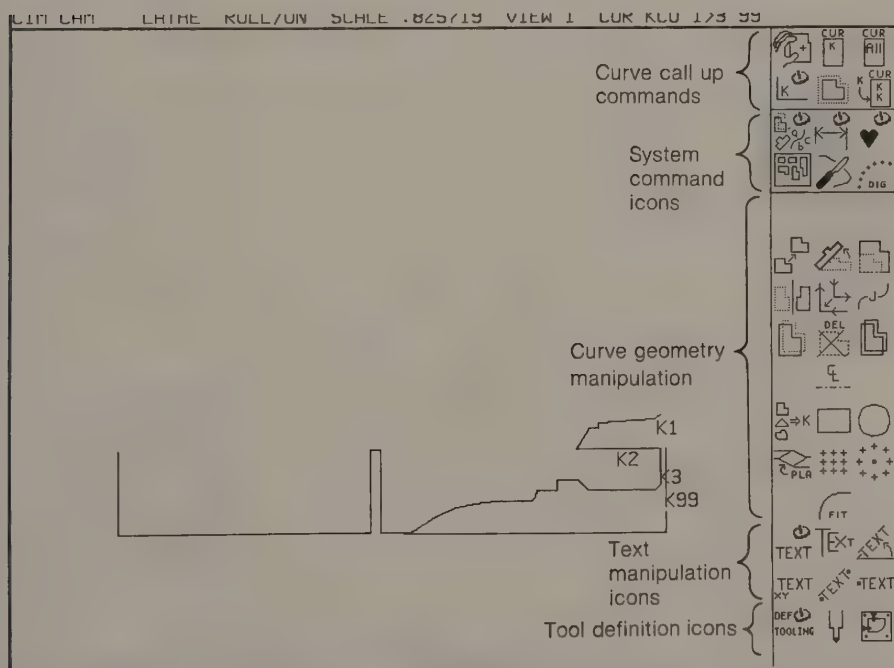


Figure 10.45 Part curve display in CIMCAM.

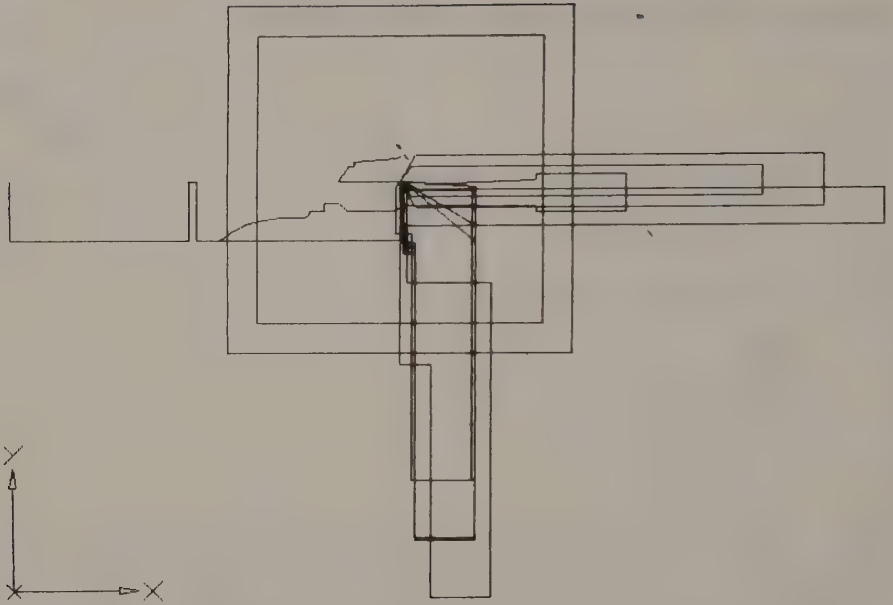


Figure 10.46 CRT display of part, turret, and tool data indicating common datum point.

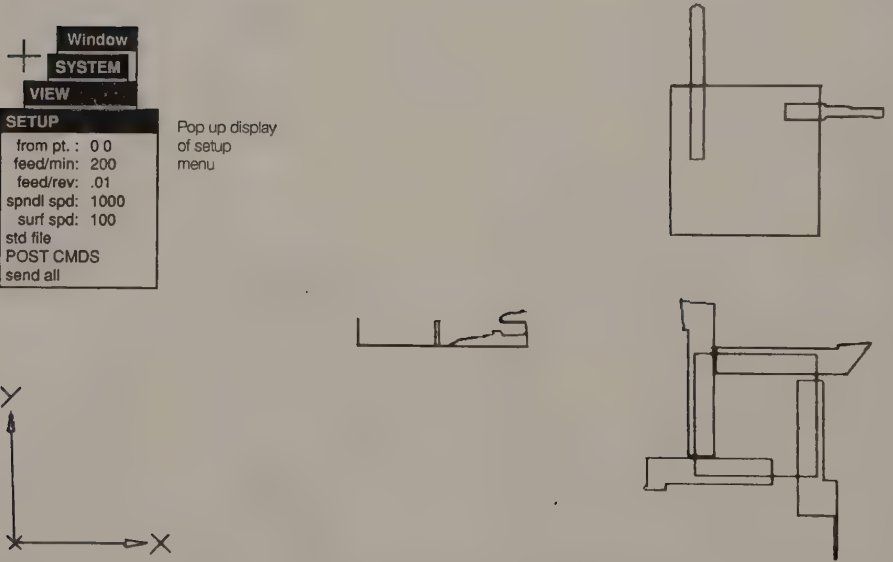


Figure 10.47 CRT display of part, turret, and tool data after completion of computer setup statement. The setup statement is completed by using the indicated pop-up menu.

to machine all spans or any percentage of a given span. This kind of freedom allows for any type of machining pass to be created.

Now that the CAD data are generated, the programmer can enter the CAM system to start the creation of machining paths. The machining system is entered by typing in the command CIMCAM. The computer system responds by asking for part data and tool library input files and a name for the output file. The CRT will then display the screen in Figure 10.44 so the type of machining to be performed (milling, lathe work, punching, or flame cutting), and inch or metric output for the machine tool can be selected. The majority of CAD/CAM systems today will allow options as to inch or metric output, and at least milling and lathe work selections. The selections on this system are completed by selecting icons with the mouse or typing the command.

Once CIMCAM is entered, the programmer can then display the machining curves or the machine's tool curves or both as in Figures 10.45 and 10.46. Using the top six icons the programmer can show and close-up on just one curve or show all curves at once. Figure 10.47 indicates the programmer's next task of properly orienting the part and machine tool turrets of tools. Figure 10.47 shows this operation completed. The programmer could then use the system to dimension Figure 10.48 and give it to the machine operator as setup instructions.

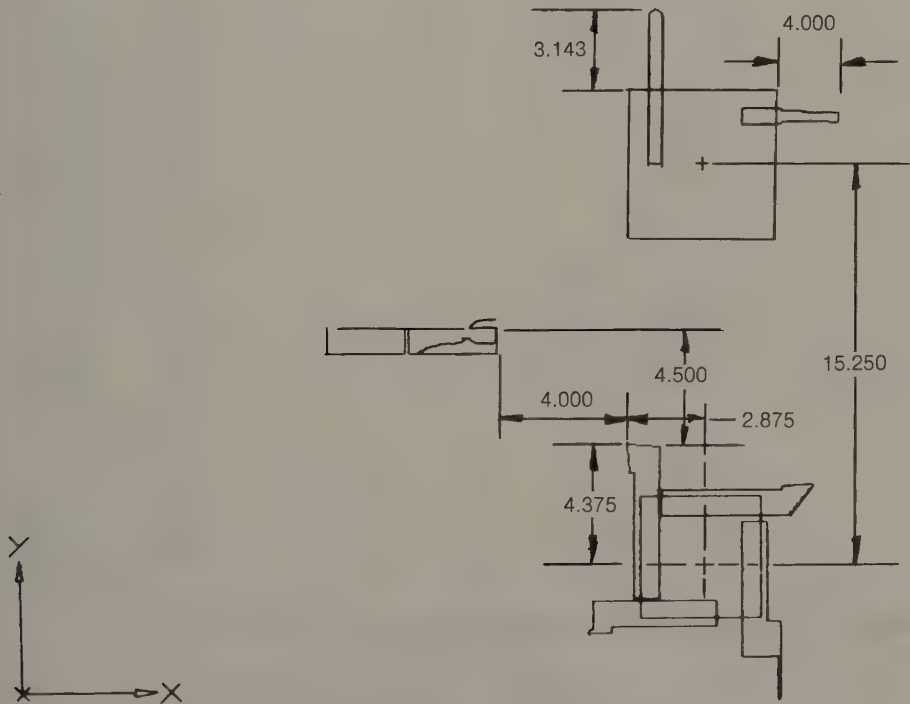


Figure 10.48 CRT display of part, turret, and tool setup dimensions.

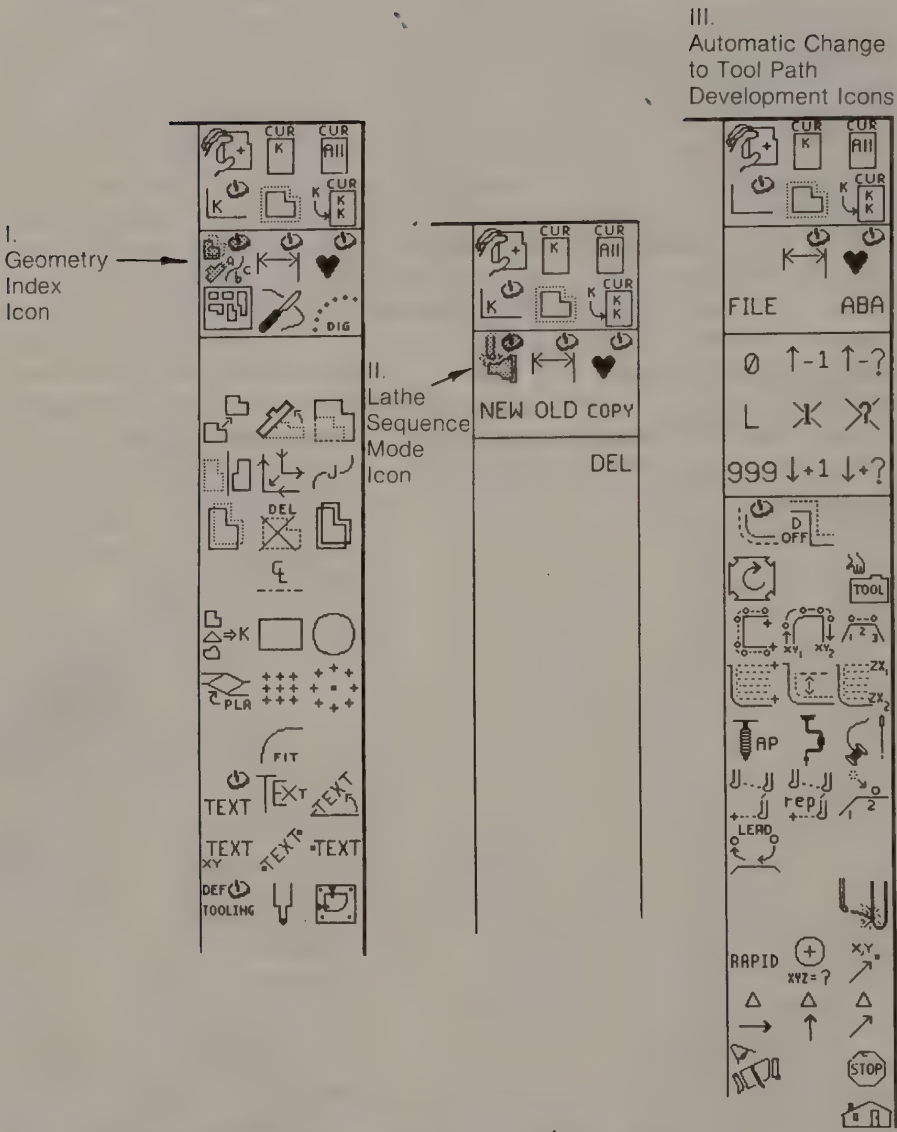


Figure 10.49 Icon commands for entering tool path generation mode.

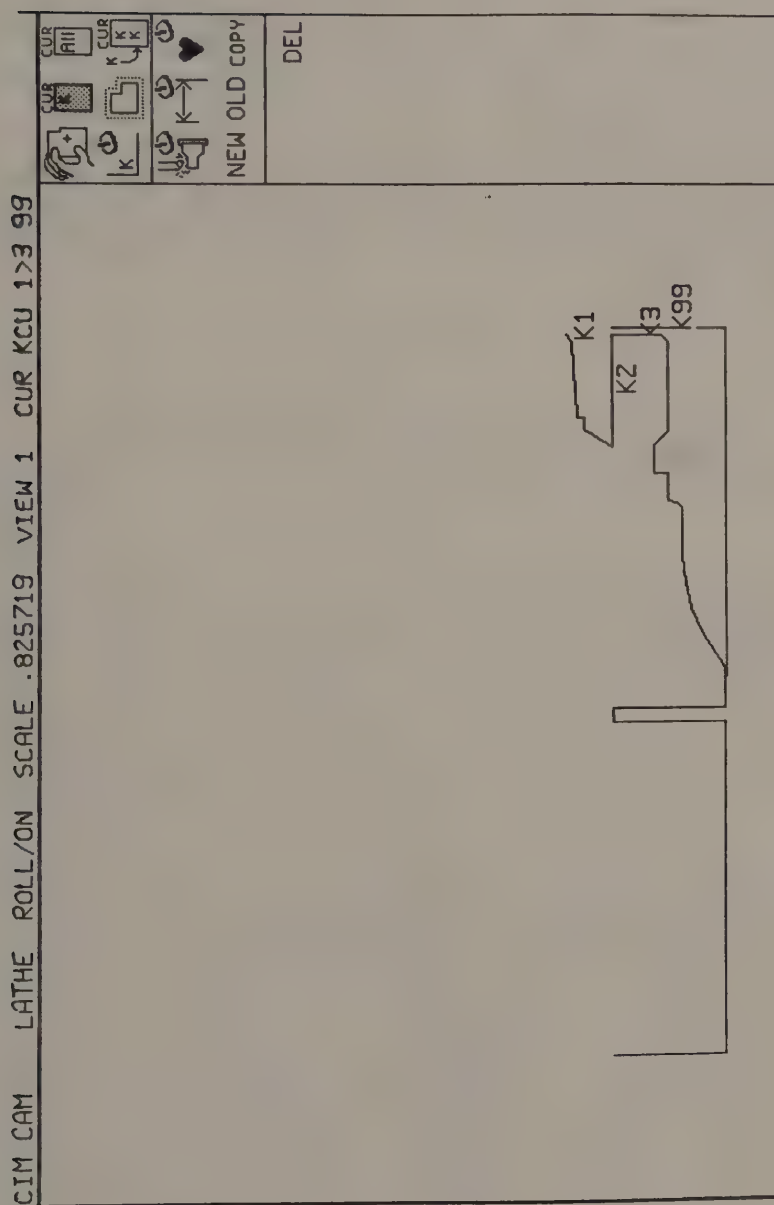


Figure 10.50 Screen display showing part curves activated by operator.

The task to perform now is to create the machining-paths that you would like the computer to create for you in the machine tools language. In order to do this, the program must enter the machining sequence mode by activating the geometry index icon to switch to the sequence or operation mode index icon display, Figure 10.49. With the sequence mode icons active, the programmer selects a new operation and enters a operation number, automatically going into the machining mode.

With tool path generation icons active, the programmer then goes step by step through the process allowing the computer system to generate the needed tool path. Figures 10.50–10.61 and computer program (Fig. 10.62) show the computer commands and program statements generated and the graphics commands used to do so.

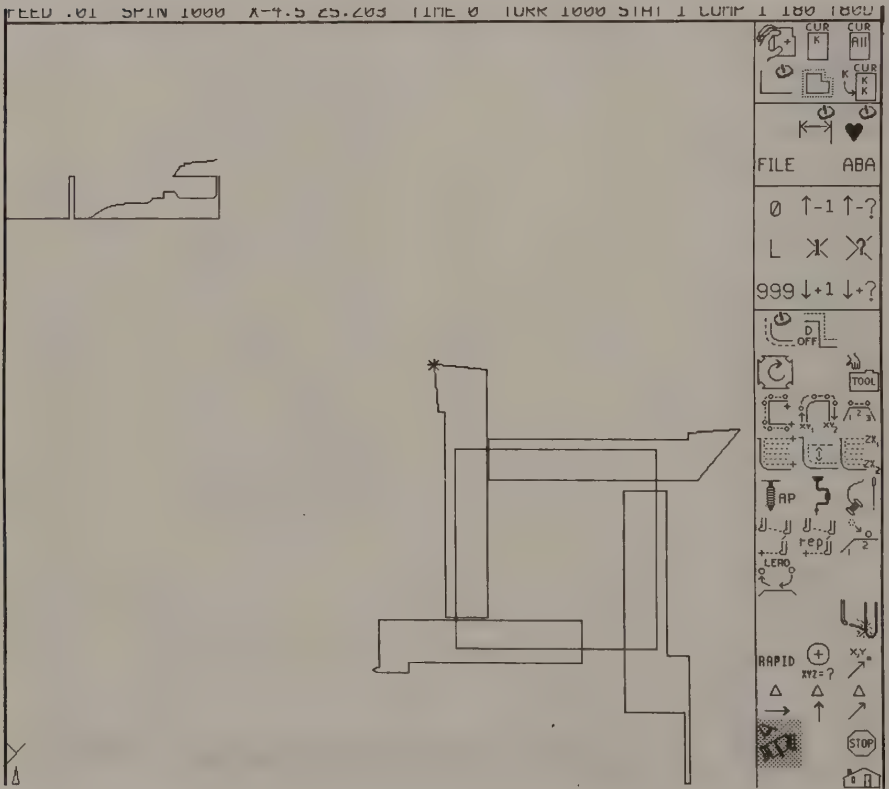


Figure 10.51 Screen display after the setup statement (Fig. 10.47) in pop-up menu has been completed. Turret display icon has been activated.

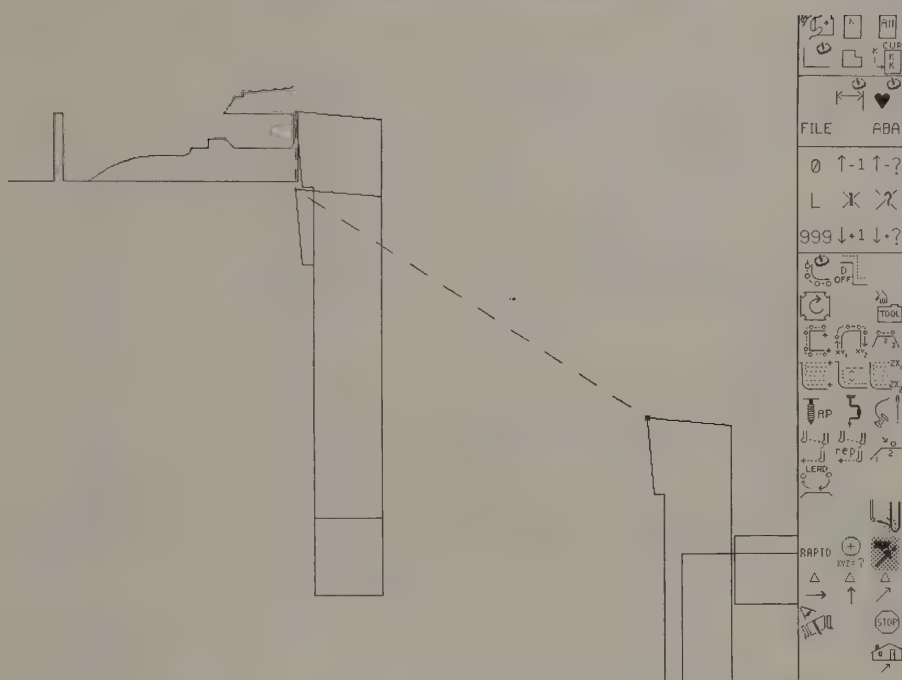


Figure 10.52 Interactive CRT display of facing operation and computer icons. (See Figure 10.62 for computer statements generated, GOTO commands—sequence 10, lines 11 and 13.).

 Icon used to set parameters for rough cutting part.

 Single GO CLEAR icon command used to produce the cut paths shown.

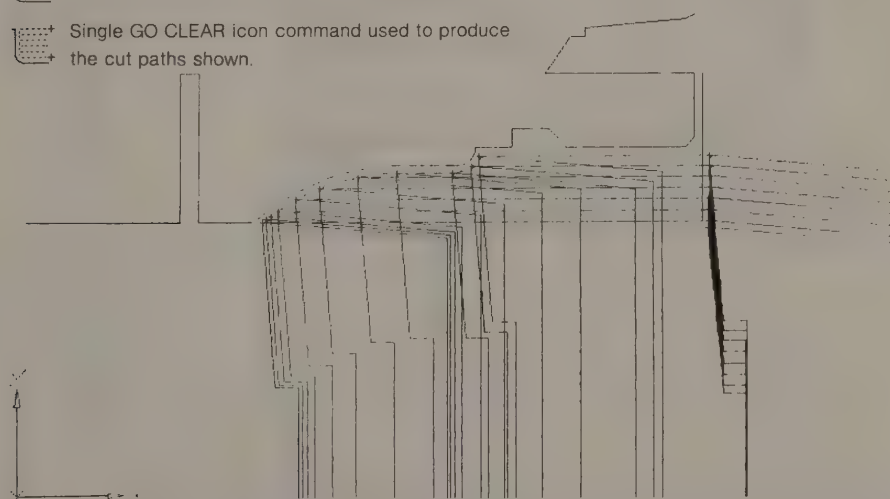


Figure 10.53 Interactive CRT display of computer-generated roughing passes. (See Figure 10.62 for computer statements generated by GO CLEAR command—sequence 10, line 23.).

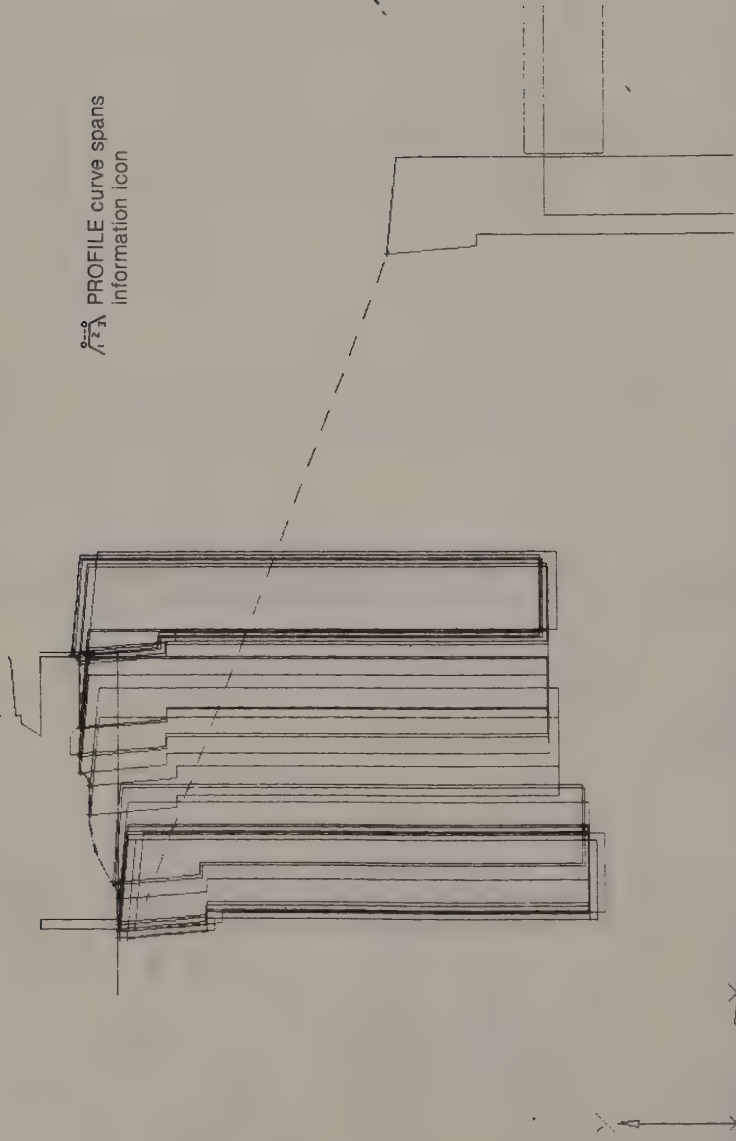


Figure 10.54 Interactive CRT display of computer-generated semifinish profile pass. (See Figure 10.62 for computer statements generated by profile command—sequence 10, lines 26, 27, and 28.).

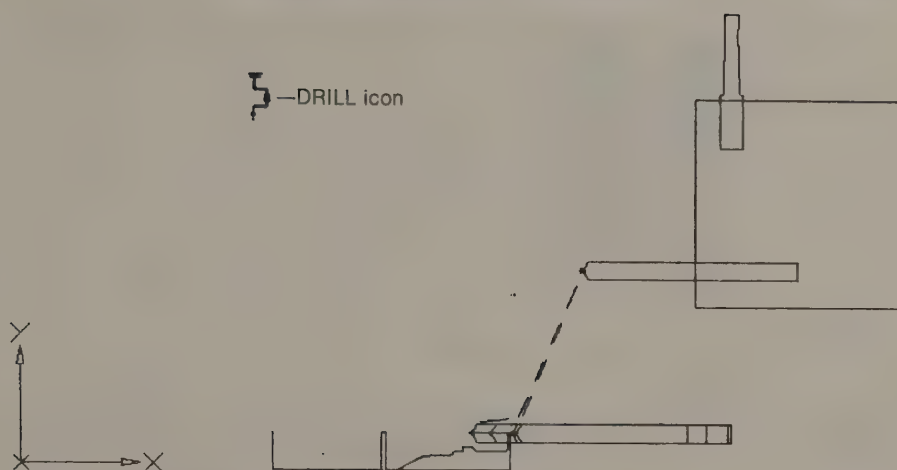


Figure 10.55 Interactive CRT display of computer-generated drilling cycle. (See Figure 10.62 for computer statements generated by the drilling command—sequence 20, line 11.)

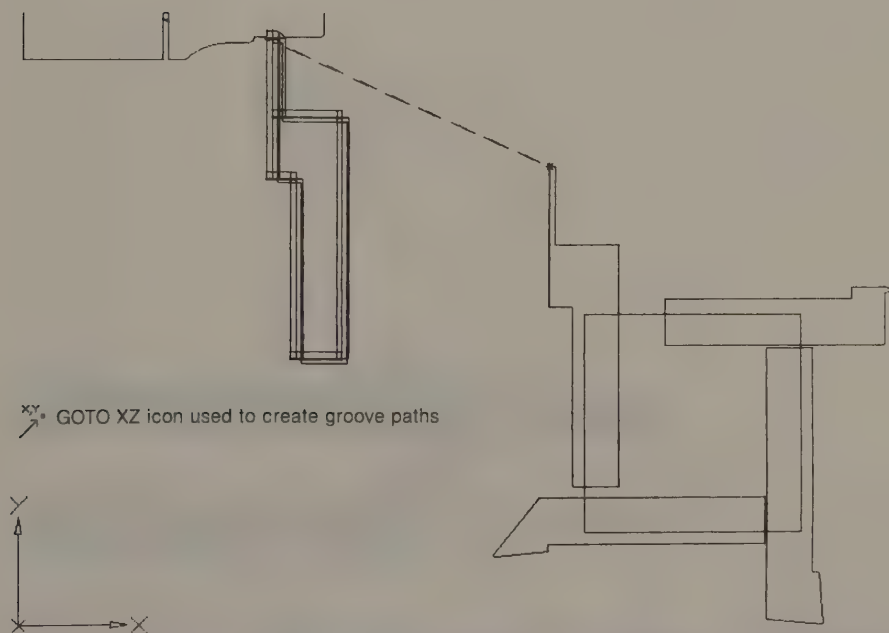
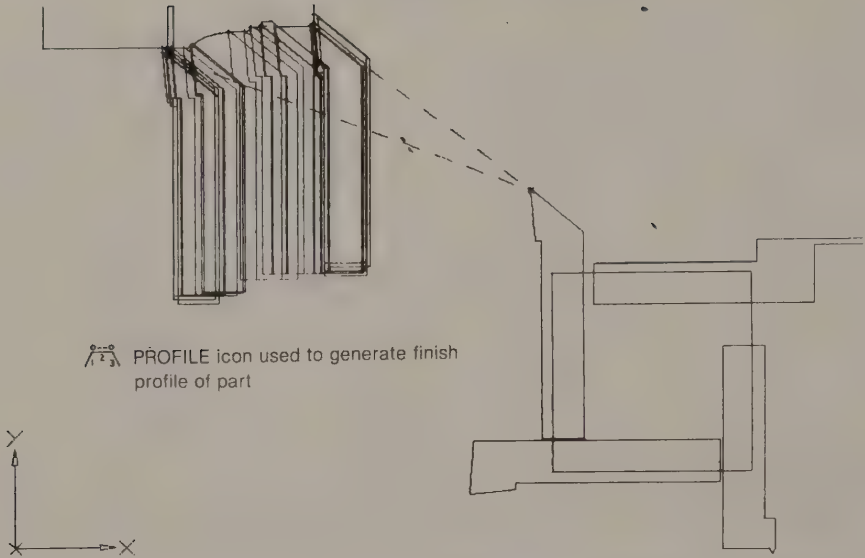
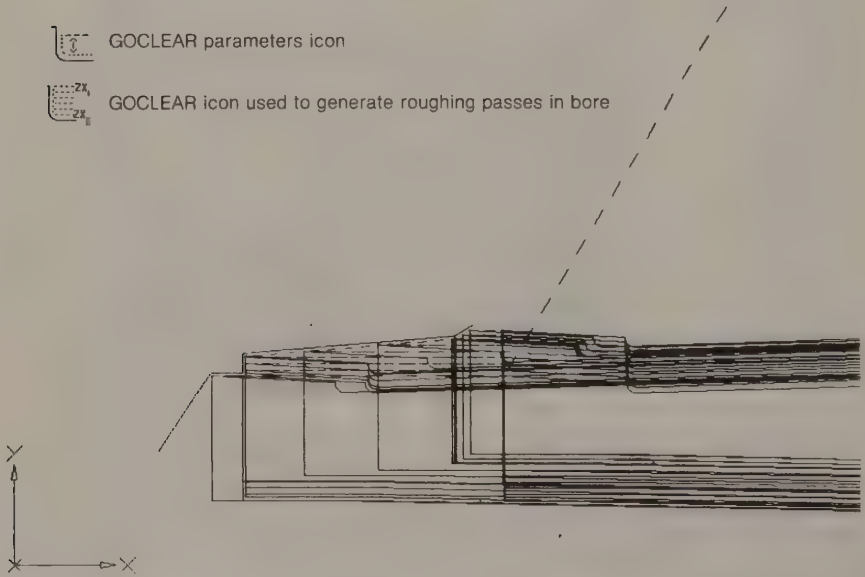


Figure 10.56 Interactive CRT display of computer-generated grooving operation. (See Figure 10.62 for computer statements generated by the GOTO command—sequence 30, lines 12, 17, and 22.)



 PROFILE icon used to generate finish profile of part

Figure 10.57 Interactive display of computer-generated finish profile pass on part. (See Figure 10.62 for computer statements generated by the PROFILE command—sequence 40, lines 15, 16, and 17.).



 GOCLEAR parameters icon

 GOCLEAR icon used to generate roughing passes in bore

Figure 10.58 Close-up interactive display of computer-generated rough boring of tapered hole. (See Figure 10.62 for computer statements generated by the GOCLEAR command—sequence 50, lines 11 and 12.).

 PROFILE icon used to generate finish bore

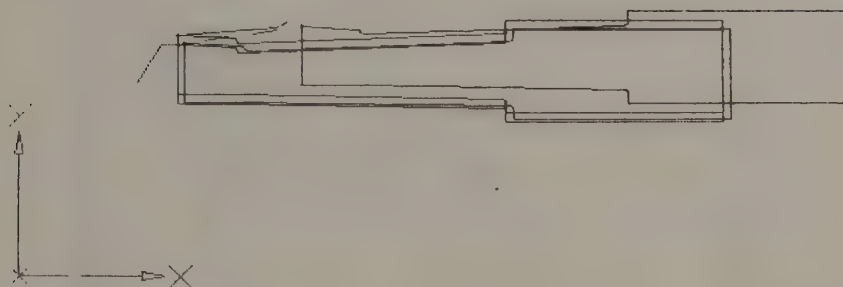


Figure 10.59 Close-up interactive display of computer-generated finish profile cut on bore. (See Figure 10.62 for computer statement generated by the PROFILE command—sequence 50, line 21.).

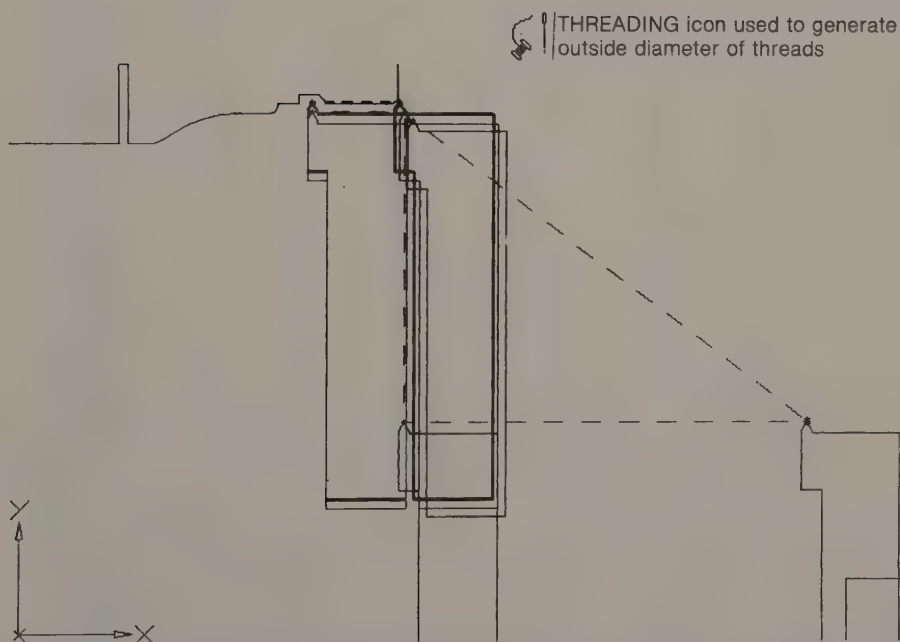


Figure 10.60 Interactive CRT display of computer-generated threading operation. (See Figure 10.62 for computer statement generated by THREADING command—sequence 60, line 12.).

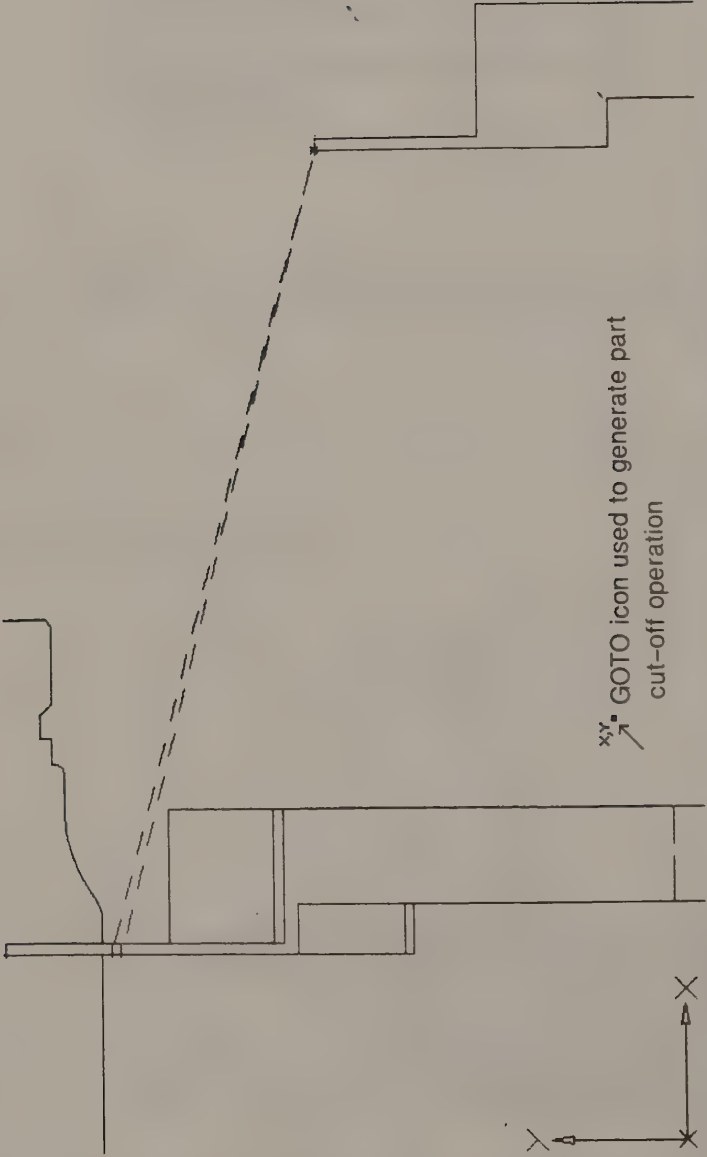


Figure 10.61 Interactive CRT display of computer-generated part cut-off "parting" operation. (See Figure 10.62 for computer statements generated by GOTO XZ command—sequence 70, lines 11 and 13.).

DATE/042190 TIME/14:44:54 CIM CAM VERSION 4.6

\$GNC/

LATHE/

KCU/1

SPA/0,0.405762

SPA/-0.060727,0.370702

SPA/-0.75,0.313262

SPA/-0.75,0.25

SPA/-0.853317,0.25

SPA/-1.021944,0

KCU/2

SPA/0,0

SPA/-1.021944,0

KCU/3

SPA/0,0

SPA/0,-0.438

SPA/-0.062,-0.5

SPA/-0.875,-0.5

SPA/-1,-0.375

SPA/-1.25,-0.375

SPA/-1.25,-0.5

SPA/-1.5,-0.5

SPA/-1.5,-0.5,-0.414214

SPA/-1.625,-0.625

SPA/-2,-0.625,0.162278

SPA/-2.9,-0.925

Part Geometry Section



Lathe Operation Icon Generated

Computer definition of drilled
and bored hole geometry

K1

Computer definition of part
centerline

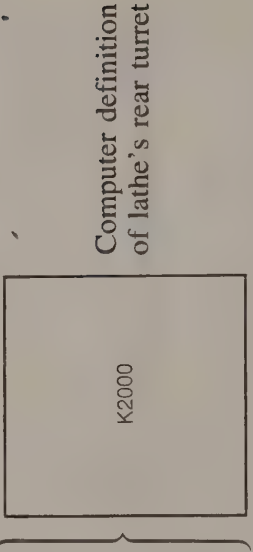
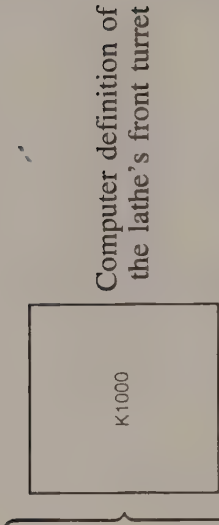
K2

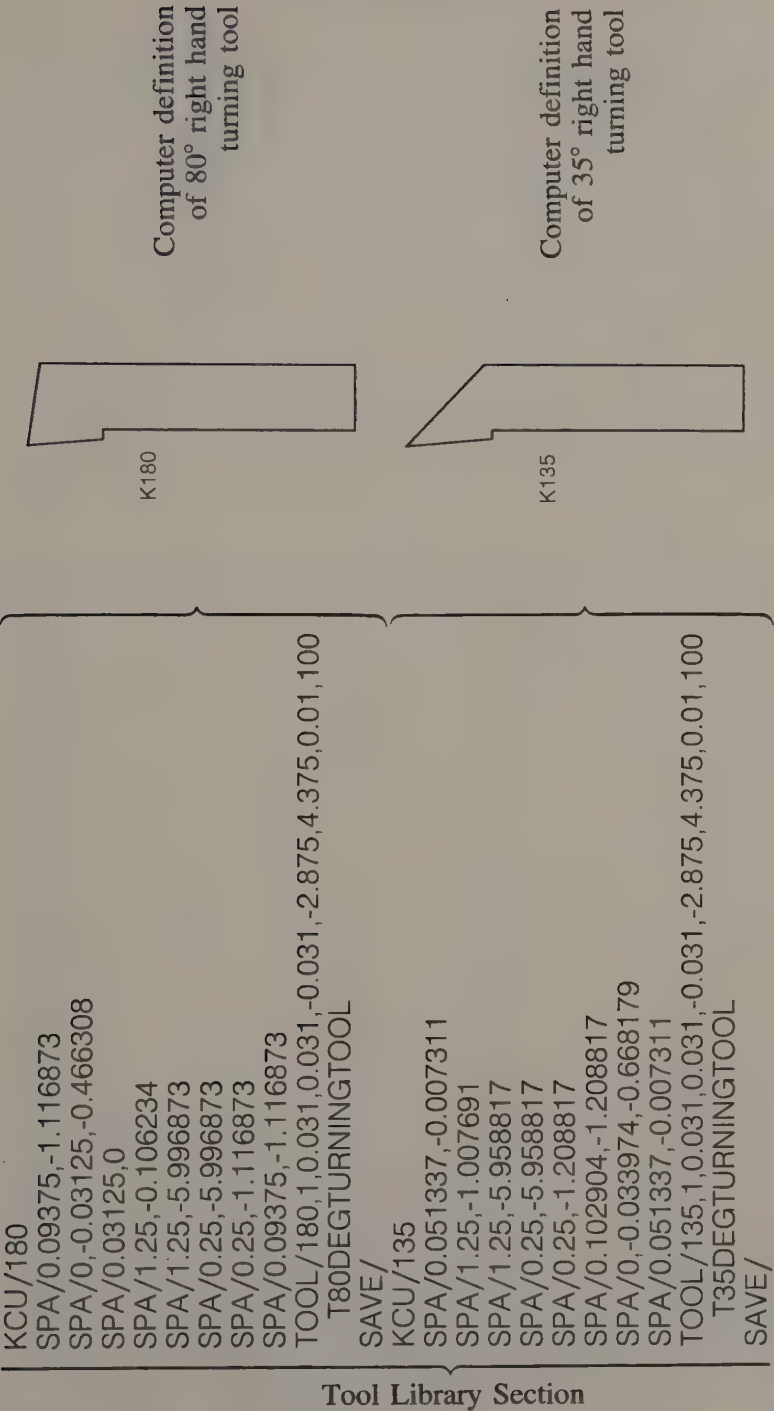
Computer definition of outside
contour of part

K3

Figure 10.62 Computer statements generated by the CAD/CAM system from which the CNC program will be created.

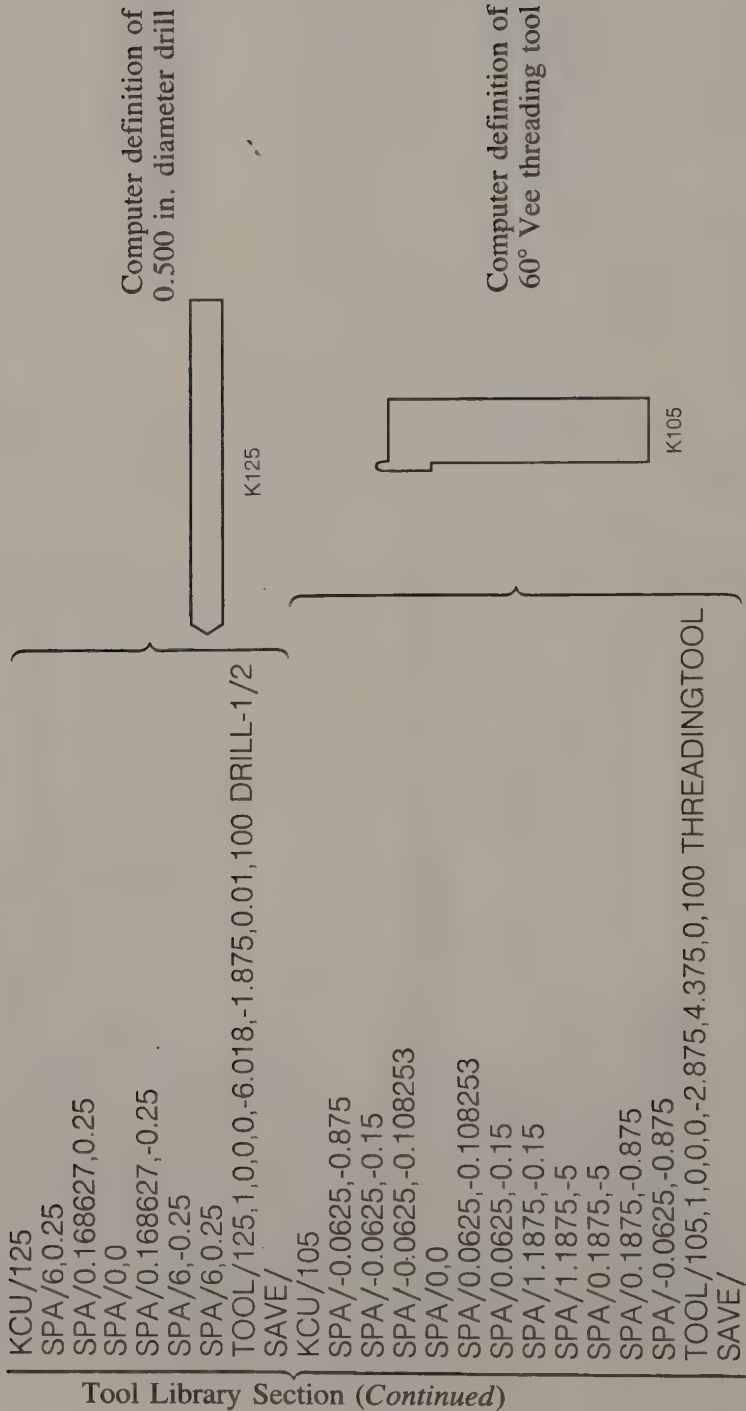
```
SPA/-2.933333,-0.95,-0.162278
SPA/-3.083333,-1
SPA/-3.375,-1
SPA/-3.375,0
SPA/-3.5,0
SPA/-3.5,-1
SPA/-6.5,-1
SPA/-6.5,0
KCU/1000
SPA/2.375,2.375
SPA/-2.375,2.375
SPA/-2.375,-2.375
SPA/2.375,2.375
SPA/2.375,-2.375
TURRET/1000,4,4,1,1,90,200,1,-99,99,-99,99 FRONTTURRET
SAVE/
KCU/2000
SPA/2.875,2.875
SPA/-2.875,2.875
SPA/-2.875,-2.875
SPA/2.875,-2.875
SPA/2.875,2.875
TURRET/2000,4,4,1,1,90,200,2,-99,99,-99,99 REARTURRET
SAVE/
```





Tool Library Section

Figure 10.62 (Continued)



Tool Library Section (Continued)

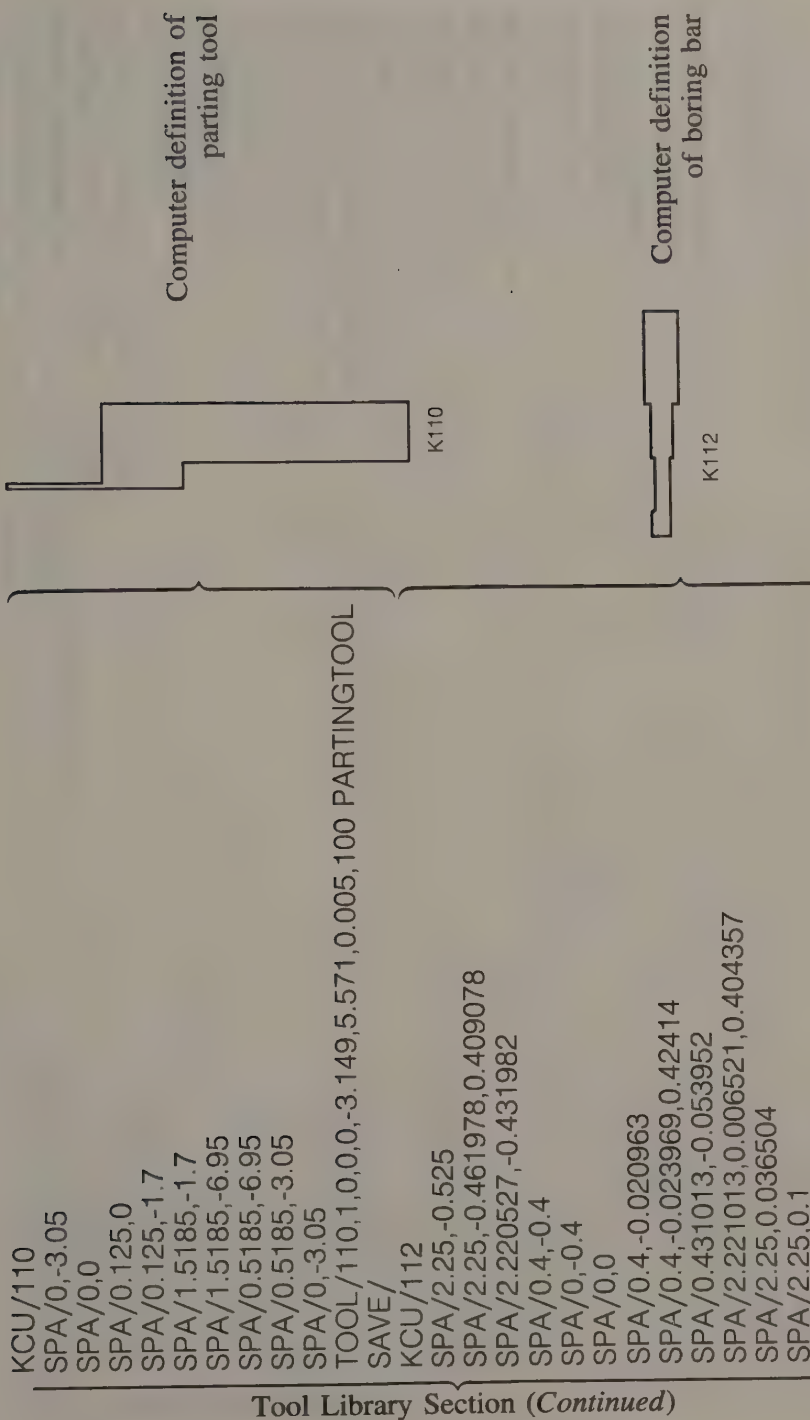


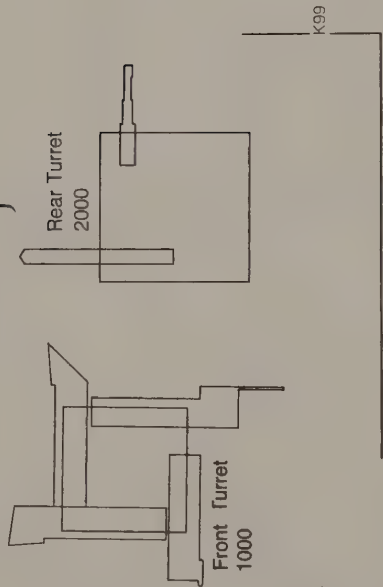
Figure 10.62 (Continued)

```

SPA/3.75,0.1
SPA/3.75,-9.525
SPA/2.25,-0.525
TOOL/112,1,0,0,-5.278,-1.663,0.01,100 BORINGBAR.5
SAVE/
SLIDE/1000,2000,0,15.25
STATION/1000,1,180
STATION/1000,2,135
STATION/1000,3,110
STATION/1000,4,105
STATION/2000,1,125
STATION/2000,2,112
KCU/99
SPA/0.06,0,0
SPA/0.06,-1.003005,0
SPA/-3.095137,-1.003005,0
NEWSEQ/10
I1 INDEX/2000,1,0
I2 FROM/2.06,4.5
I3 SPIN/1000,0
I4 FROM/2.06,4.5
I5 FPM/200
I6 FPR/0.01
I7 SPIN/1000,0
I8 INDEX/2000,4,0
I9 INDEX/1000,1,1
I10 RAPID/
I11 GOTO/0.03,-1.112

```

Start Facing Operation



Computer commands
to place needed tools
in library into proper
machine turret
positions

Computer definition
of material stock
curves



Index rear turret to clearance
Enter setup position of tools to part
Set spindle speed

Set rapid feedrate
Set cutting feedrate

Index rear turret to position
Index front turret to position
Set rapid motion
Move to clearance position
to start cut (Fig. 10.52)

Rapid Movement Icon RAPID
GO TO XZ
Position Icon



Create file of machining pathes
Machine time is 2.64 minutes

Index rear turret
Enter set-up position of tool to part
Enter machine spindle speed

Set rapid feed rate
Set cutting feed rate
Enter new spindle speed
Set rapid movement
Turn coolant on

Rapid tool to end of part
Drill part command (Fig. 10.55)
Turn coolant off

Set rapid movement
Clear tool from part
Set rapid movement
Return turret to tool change position
Create file of drilling tool pathes
Machine time for drilling operation

Enter set up tool
Enter set position of tool to part

Set up information



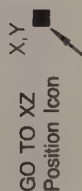
Drilling Operation (Fig. 55)	Grooving Operation
<pre> FILE/ M/C TIME 2.64 NEWSEQ/20 !1 INDEX/2000,1,5 !2 FROM/2.06,4.5 !3 SPIN/1000,0 !4 FROM/2.06,4.5 !5 FPM/200 !6 FPR/0.01 !7 SPIN/500,0 !8 RAPID/ !9 COOL/3 !10 GOTO/0.1,0 !11 DRILL/0.5,0.1,-1.022,2 !12 COOL/2 !13 RAPID/ !14 GODEL/0.1 !15 RAPID/ !16 GOHOME/ </pre>	<pre> FILE/ M/C TIME 0.29 NEWSEQ/30 !1 INDEX/2000,1,0 !2 FROM/2.06,4.5 !3 SPIN/1000,0 !4 FROM/2.06,4.5 !5 FPM/200 </pre>

Index rear turret to clearance position
 Index front turret to parting tool
 Enter proper cutting rpm
 Set rapid movement
 Rapid to start grooving operation
 Turn coolant on

Dwell at end of cut
 Move to clear part
 Set rapid movement
 Rapid to next groove point
 Cut groove area to depth
 (Fig. 10.56) GOTO XZ

Position Icon
 Dwell at end of cut
 Move to clear part
 Set rapid movement
 Rapid to next groove point

Dwell at end of cut
 Move to clear part
 Set rapid movement
 Rapid clear of part
 Set rapid movement
 Turn coolant off
 Return turret to tool change position



```

!6 INDEX/2000,3,0
!7 INDEX/1000,3,3
!8 SPIN/300,0
!9 RAPID/
!10 GOTO/-1.25,-0.55
!11 COOL/3
!12 GOTO/-1.25,-0.375
!13 DWELL/2
!14 GODEL/0,-0.15
!15 RAPID/
!16 GOTO/-1.125,-0.55
!17 GOTO/-1.125,-0.375
!18 DWELL/2
!19 GODEL/0,-0.15
!20 RAPID/
!21 GOTO/-0.975,-0.525
!22 GOTO/-1.125,-0.375
!23 DWELL/2
!24 GODEL/0,-0.15
!25 RAPID/
!26 GODEL/0.1,-0.1
!27 RAPID/
!28 COOL/2
!29 GOHOME/
  
```

(Fig. 56)

Figure 10.62 (Continued)

Create file of machining pathes
Machine time for grooving
is 0.17 minute

FILE/
_M/C TIME 0.17

```

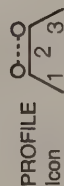
--NEWSEQ/40
!1 INDEX/2000,1,0
!2 FROM/2.06,4.5
!3 SPIN/1000,0
!4 FROM/2.06,4.5
!5 FPM/200
!6 FPR/0.01
!7 INDEX/2000,3,0
!8 INDEX/1000,2,2
!9 RAPID/
!10 GOTO/0,-0.55
!11 COOL/3
!12 GOTO/0,-0.22
!13 RAPID/
!14 GODEL/0.1,-0.1
!15 PROF/3,0,2,0,0,4,0,0
!16 PROF/3,0,-1.25,-0.5,-3.437,-1
!17 PROF/3,0,17,0,0,17,0,0.1
!18 GODEL/0,-0.1
!19 RAPID/
!20 COOL/2
!21 GODEL/0.1,-0.1

```

Finish Part Profile

Setup information

Index rear turret to
clearance position
Index front turret to
35° finish turning tool
Set rapid movement
Position tool for finish face
Turn coolant on
Finish face part
Set rapid movement
Retract clear of part to
prepare for finish profile.



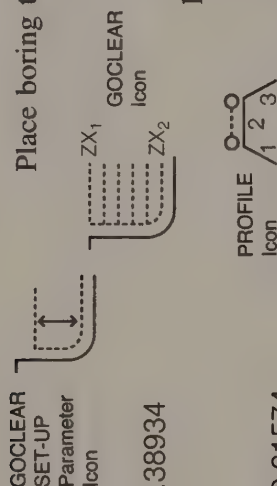
Finish turn outside profile of part
(Fig. 10.57)
Face cut off part surface
Set rapid movement
Turn coolant off
Retract from part

Set rapid movement
Return turret to tool change position
Create file of machining pathes
Machine time for finishing
the outside profile is 0.51 minute

Set up information

Index boring bar into position
Set rapid movement
Place boring tool in front of drilled hole
Turn coolant on
Set roughing tool
path parameters
Roughing tool
pathes created (Fig. 10.58)

Set rapid movement
Position for finish face
Finish face c'bore
Retract from counterbore face
Set rapid movement
Retract from bore



```

!22 RAPID/
!23 G0H0ME/
FILE/
  _M/C TIME 0.51
--NEWSEQ/50
!1 INDEX/2000,1,0
!2 FROM/2.06,4.5
!3 SPIN/1000,0
!4 FROM/2.06,4.5
!5 FPM/200
!6 FPR/0.01
!7 INDEX/2000,2,6
!8 RAPID/
!9 GOTO/0.1,0.24
!10 COOL/3
!11 STEP/0.02,90,180,-45,1,1,-1
!12 GOCLR/1,0.01,0.1,0.24,0.09626,0.38934
!13 RAPID/
!14 GOTO/0.25,-0.75
!15 PROF/1,0,-0.75104,0.2229,-0.75,0.31574
!16 GODEL/0.05,-0.05
!17 RAPID/
!18 GOZX/0.05704,0.31949,1
  
```

Figure 10.62 (Continued)

Rough and Finish Bore

```
!19 RAPID/  
!20 GOZX/0.05591,0.40951,-1  
!21 PROF/1,0,0.00043,0.40951,-0.75,031574  
!22 GODEL/0.05,-0.05  
!23 RAPID/  
!24 GOZX/0.24252,0.21574,-1  
!25 COOL/2  
!26 RAPID/  
!27 GOHOME/  
FILE/
```

```
____M/C TIME 0.67
```

Threading Operation

```
—NEWSEQ/60  
!1 INDEX/2000,1,0  
!2 FROM/2.06,4.5  
!3 SPIN/1000,0  
!4 FROM/2.06,4.5  
!5 FPM/200  
!6 FPR/0.01  
!7 INDEX/2000,3,0
```

```
!8 INDEX/1000,4,4
```



Set rapid movement
Position tool to finish profile bore
Finish profile bore (see Fig. 10.59)
Move clear of side of bore
Set rapid movement
Retract from bore
Turn coolant off
Set rapid movement
Return turret to tool change position
Create file of machining pathes
Machine time for roughing
and finishing bore is
0.67 minute

Set up information,

Index rear turret to
clearance position

Index front turret to
threading tool

Set spindle speed
 Turn coolant on
 Select slash delete option on
 Threading tool paths
 "O.D." (See Fig. 10.60)
 Turn slash delete off
 Turn coolant off
 Set rapid movement
 Retract tool clear of part
 Set rapid movement
 Return turret to tool change position
 Create file of machining paths
 Machine time for producing threads
 is 0.12 minute



Set up information

Index rear turret to clearance position
 Turn slash delete option on
 Index front turret to part off tool
 Set spindle speed
 Set rapid movement

```

!9  SPIN/300,0
!10 COOL/3
!11 OPSKIP/
!12 THREAD/20,1,0,1,-0.5,-1.07,
    -0.5,0.035,30,3,2,1,0.1
!13 OPSKIP/
!14 COOL/2
!15 RAPID/
!16 GODEL/0.1,-0.1
!17 RAPID/
!18 GOHOME/
FILE/
  _M/C TIME 0.12
NEWSEQ/70
!1  INDEX/2000,1,0
!2  FROM/2.06,4.5
!3  SPIN/1000,0
!4  FROM/2.06,4.5
!5  FPM/200
!6  INDEX/2000,3,0
!7  OPSKIP/
!8  INDEX/1000,3,3
!9  SPIN/300,0
!10 RAPID/
  
```

Part Cutoff Operation

Figure 10.62 (Continued)

```

!11 GOTO/-3.5,-1.1
!12 COOL/3
!13 GOTO/-3.5,0
!14 RAPID/
!15 GOTO/-3.5,-1.2
!16 RAPID/
!17 COOL/2
!18 GOHOME/
!19 OPSKIP/
FILE/
M/C TIME 0.1
EOF/EOF/

```

Figure 10.62 (Concluded)

Position tool to part
(see Fig. 10.61)

Turn coolant on
Cut part off
Set rapid movement
Retract from part
Set rapid movement
Turn coolant off

Return turret to tool change position
Turn slash delete option off
Create file of machining pathes
Machine time for cutting part off
End of file

GOTO
Icon

X,Y



The machining path commands when finished are placed in one master file by filing out of the sequence mode of operation (see Figure 10.63). The programmer now has a file of computer commands that will generate a CNC machine program. The machine program is created by running the computer program through a postprocessor program in the computer. The postprocessor program is a translator that converts the computer program to a CNC program for the particular machine you will be using. There is normally a postprocessor file for each different machine you have. This process is very simply completed by activating first the program to convert and then the postprocessor required, through typed in commands. The output of our example program for the screw jack can be seen in Figure 10.64. It is also indicated on the CNC program what

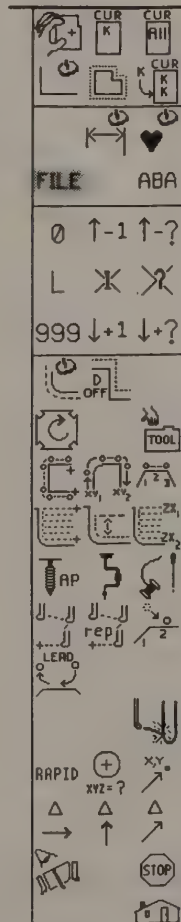


Figure 10.63 FILE icon command used to exit the tool path generation mode creating a master computer machining file.

Oct 1 17:35 1990 tc1 Page 1
###SHEAB_1
###DATE 09/20/90 TIME 19:29 HRS

PARTNO/Jack Screw 87231-2
CNC Machine Commands

Date and time of run—File name
Postprocessor name
Date and time of computer
source file Figure 10.62
Part identification
Computer statement
Generating CNC Command

Rewind Stop Code

Setup SEQ 10—Lines 1–7

Line 8

Line 9—80° right-hand turning tool

Lines 1–7

Line 7

Lines 10 & 11 }
Line 12 }
Line 13 } Rough face part
Lines 14 & 15 }

Lines 16 & 17

Sequence #10 Rough—Face & Turn Figures 10.52–10.54

%
N0010G70
N0020G90
N0030G00T00
N0040T0700
N0050T0101
N0060G95
N0070G92X4.5Z5.203
N0080G97S1000M03
N0090G00X1.11Z.03
N0100M08
N0110G01X-.03F.01
N0120G00X.07Z.13
N0130X1.05Z.1
N0140G01Z-3.625
N0150X1.15
N0160G00X1.05Z.1
N0170X1.
N0180X.925
N0190G01Z-2.8354
N0200X.9588Z-2.8824
N0210X.9838Z-2.9157
N0220X1.Z-2.9394
N0230G00Z.1
N0240X.85
N0250G01Z-2.7137
N0260G03X.925Z-2.8354I1.244K.6827
N0270G00Z.1011
N0280X.775
N0290G01Z-2.5543
N0300G03X.85Z-2.7137I1.319K.5233
N0310G00Z.1022
N0320X.7
N0330G01Z-2.2962
N0340G03X.775Z-2.5543I1.394K.2652
N0350G00Z.1033
N0360X.6319
N0370G01Z-1.53

Lines 18–25—
Rough turn part

Figure 10.64 Postprocessor output of CNC data for jack screw example to run on a Sheldon lathe.

N0380G02X.675Z-1.656I.1629K.126
 N0390G01Z-2.031
 N0400G03X.7Z-2.2962I1.419K0.
 N0410G00Z.1044
 N0420X.5639
 N0430G01Z-1.4732
 N0440G02X.6319Z-1.53I.0949K.1829
 N0450G00Z.1054
 N0460X.4176Z.061
 N0470G01X.4073Z.02
 N0480G02X.4431Z.0051I.0003K.051
 N0490G01X.5051Z-.0569
 N0500G02X.52Z-.093I.0361K.0361
 N0510G01Z-.906
 N0520G02X.5161Z-.9255I.051K0.
 N0530G01X.5051Z-1.2449
 N0540G02X.52Z-1.281I.0361K.0361
 N0550G01Z-1.4876
 N0560G02X.645Z-1.656I.051K.1684
 N0570G01Z-2.031
 N0580G03X.9348Z-2.9004I1.449K0.
 N0590G01X.9598Z-2.9337
 N0600G02X1.02Z-3.1143I.2408K.1806
 N0610G01Z-3.406
 N0620G02X1.0051Z-3.4421I.051K0.
 N0630G01Z-3.4949
 N0640G02X1.02Z-3.531I.0361K.0361
 N0650G01Z-3.6107
 N0660X1.12
 N0670G00X1.22Z-3.5107
 N0680M09
 N0690X4.5Z5.203T0000

Lines 25–28—
 Semifinish part profile

Line 29—Face clear of part
 Lines 30 & 31—Clear part
 Line 32—Turn coolant off
 Lines 33 & 34—
 Send turret to tool change

—N0700T0505

SEQ. 20—Line 1—
 Index rear turret to .500 drill

N0710G95
 N0720G92X-4.5Z2.06
 N0730G97S500M03
 N0740G00X0.Z.1M08
 N0750G01Z-.4905F.01
 N0760G00Z.1
 N00770Z-.461
 N0780G01Z-1.022
 N0790G00Z.1
 N0800M09
 N0810Z.2
 N0820X-4.5Z2.06T0000

Lines 2–7—Setup information

Lines 8–10—position drill to part

Line 11—Drill part—0.500 in. dia. hole

Line 12—Turn coolant off
 Lines 13 & 14—Clear tool from part
 Lines 15 & 16—Send turret to tool change

Figure 10.64 (Continued)

Sequence #30 Grooving Operation Figure 10.56

—N0830T0303

N0840G95
N0850G92X3.304Z4.929
N0860G97S300M03
N0870G00X.55Z-1.25
N0880M08
N0890G94
N0900G01X.375F200.
N0910G04F2.
N0920X.525
N0930G00X.55Z-1.125
N0940G01X.375
N0950G04F2.
N0960X.525
N0970G00Z-.975
N0980G01X.375Z-1.125
N0990G04F2.
N1000X.525
N1010G00X.625Z-1.025

N1020M09
N1030X3.304Z4.929T0000

Sequence #40 Finish Turning Operation Figure 10.57

—N1040T0202

N1050G94
N1060G92X4.5Z5.203
N1070G97S1000M03
N1080G00X.55Z0.
N1090M08
N1100G95
N1110G01X.22F.01
N1120G00X.32Z.1
N1130G01X.4289Z-.0091
N1140X.4909Z-.0711
N1150G02X.5Z-.0931.0219K.0219
N1160G01Z-.906
N1170G02X.4909Z-.92791.031K0.
N1180G01Z-1.2591
N1190G02X.5Z-1.2811.0219K.0219
N1200G01Z-1.5031
N1210G02X.625Z-1.6561.031K.1529
N1220G01Z-2.031
N1230G03X.9188Z-2.912411.469K0.
N1240G01X.9438Z-2.9457
N1250G02X1.Z-3.11431.2248K.1686

SEQ. 30—
Line 7—Index front turret to parting tool

Lines 1–8—Setup information

Lines 9 & 10—Position groove

Line 11—Turn coolant on

Line 12—Cut groove

Line 13—Dwell at bottom of groove

Line 14—Clear part

Lines 15–24—Cut remaining groove area

Lines 25 & 26—Clear part

Line 28—Turn coolant off

Lines 27 & 29—Send turret
to tool change

SEQ. 40—Line 8—Index front turret
to 35° finish turning tool

Lines 1–7—Setup information

Lines 9 & 10—Position for finish face

Lines 11 & 12—Finish face part

Lines 13 & 14—Move clear of part

Lines 15–17—
Finish profile part

Figure 10.64 (Continued)

Sequence #50 Rough and Finish Bore Figures 10.58 and 10.59

N1260G01Z-.3.406
 N1270G02X.9909Z-.3.4279I.031K0.

N1280G01X1.Z-.3.531

N1290Z-.3.631

N1300X1.1

N1310M09

N1320G00X1.2Z-.3.531 }

N1330X4.5Z5.203T0000

—N1340T0606

N1350G95

N1360G92X-.4.712Z2.8 }

N1370G97S1000M03 }

N1380G00X-.24Z.1

N1390M08

N1400X-.26

N1410G01Z-.74F.01

N1420X-.25

N1430G02X-.24Z-.75I0.K.01

N1440G01Z-.848

N1450G00Z.1

N1460X-.28

N1470G01Z-.74

N1480X-.26

N1490G00Z.0995

N1500X-.3

N1510G01Z-.74

N1520X-.28

N1530G00Z.099

N1540X-.32

N1550G01Z-.5487

N1560X-.3041Z-.74

N1570X-.3

N1580G00Z.0985

N1590X-.34

N1600G01Z-.3087

N1610X-.32Z-.5487

N1620G00Z.098

N1630X-.36

N1640G01Z-.0687

N1650X-.34Z-.3087

N1660G00Z.0975

N1670X-.3747

N1680G01Z-.0339

N1690X-.362Z-.0557

Line 18—Face off of part

Lines 19–21—

Move clear of part/coolant off

Lines 22 & 23—

Send turret to tool change

SEQ. 50—Line 7—

Index rear turret to boring bar

Lines 1–6—Setup information

Lines 8–9—Position to drilled hole

Line 10—Coolant on

Lines 11 & 12—

Rough Bore Figure 10.58

Figure 10.64 (Continued)

N1700X-.3607Z-.0599
 N1710X-.36Z-.0687
 N1720G00Z.097
 N1730X-.3893
 N1740G01Z-.0084
 N1750X-.3747Z-.0339
 N1760G00Z.0966
 N1770X-.25Z-.751
 N1780G01Z-.75
 N1790X-.3133
 N1800Z-.7498
 N1810X-.2633Z-.6998
 N1820G00X-.3195Z.057
 N1830X-.4095Z.0559
 N1840G01X-.4058Z0.
 N1850X-.3707Z-.0607
 N1860X-.3133Z-.7498
 N1870X-.2633Z-.6998
 N1880G00X-.2157Z.2425
 N1890M09
 N1900X-4.712Z2.8T0000

Lines 14–18—
 Finish face counterbore

Lines 19–22—Finish bore Figure 10.59

Lines 23 & 24—Retract from bore
 Line 25—Coolant off
 Lines 26 & 27—
 Return turret to tool change position

SEQ. 60—Line 8—Index front turret
 to threading tool

Lines 1–9—Setup information

One thread pass

Lines 10–13—
 Threading operation
 with operation skip option

—N1910T0404

Sequence #60 Threading Operation Figure 10.60
 /N1920G95
 /N1930G92X4.5Z5.203
 /N1940G97S300M03
 /N1950G00X4.5Z.1M08
 /N1960X.6
 /N1970X.4907Z.0369
 /N1980G33Z-1.07K0.0417
 /N1990G00X.6
 /N2000Z.1
 /N2010X.4813Z.0315
 /N2020G33Z-1.07K0.0417
 /N2030G00X.6
 /N2040Z.1
 /N2050X.472Z.0261
 /N2060G33Z-1.07K0.0417
 /N2070G00X.6
 /N2080Z.1
 /N2090X.4685Z.0241
 /N2100G33Z-1.07K0.0417
 /N2110G00X.6
 /N2120Z.1
 /N2130X.465Z.0221
 /N2140G33Z-1.07K0.0417
 /N2150G00X.6
 /N2160Z.1

Figure 10.64 (Continued)


```

/N2170X.465Z.0221
/N2180G33Z-1.07K0.0417
/N2190G00X.6
/N2200Z.1
N2210M09
N2220G00X.7Z.2
N2230X4.5Z5.203T0000

```

Line 14—Coolant off
 Lines 15 & 16—Move clear of part
 Lines 17 & 18—
 Return turret to tool change position

SEQ. 70—Line 6—Index front turret
 to parting tool

Lines 1–9—Setup information

Lines 10 & 11—Position tool to part
 Line 12—Turn coolant on

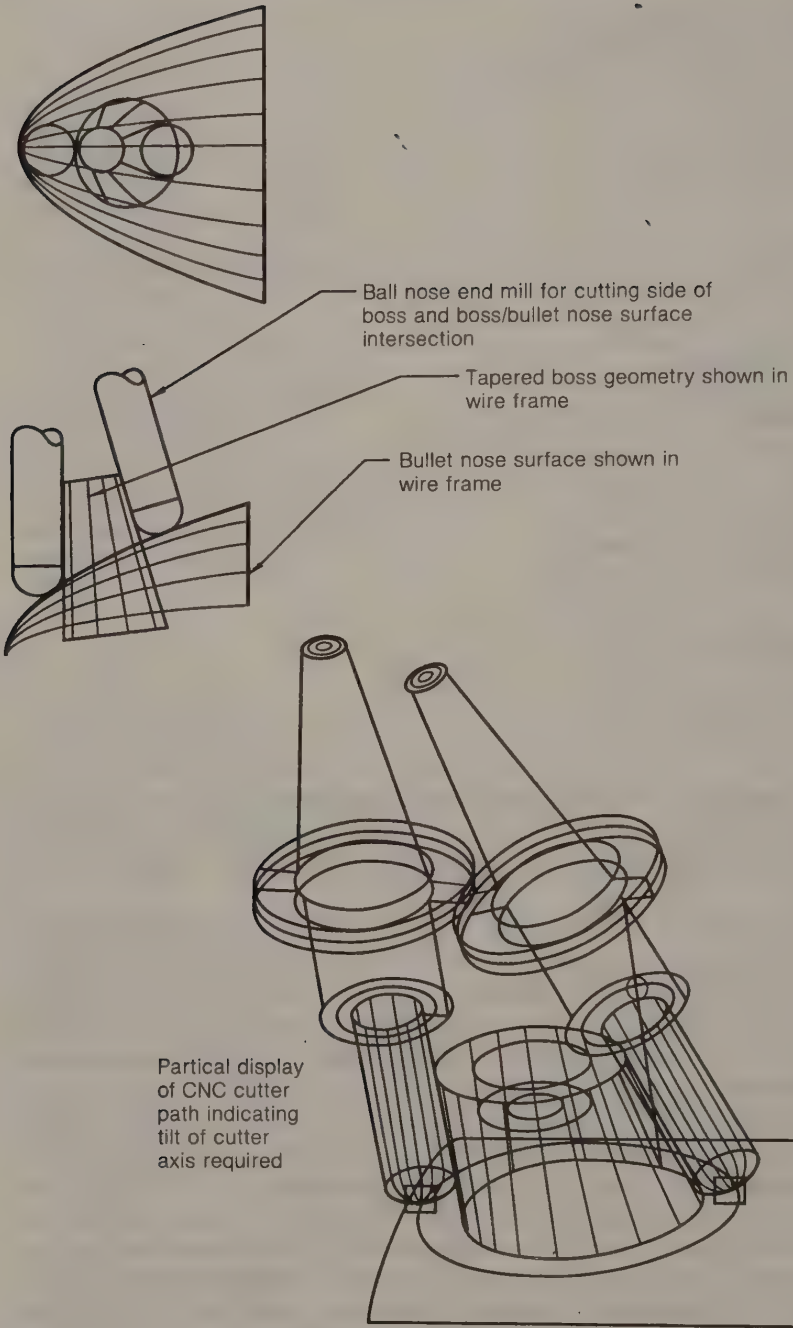
Line 13—Cut part off
 Lines 14 & 15—Retract clear of part
 Lines 16–19—Turn coolant off return
 turret to tool change position
 End of program

Figure 10.64 (Concluded)

computer statements generated what machine statements. By using a few more commands to the CAD/CAM system, we could transfer the information to the machine tool by electronic means or punch a tape and take it to the machine control.

You have now seen the basic powers of a CAD/CAM system, but they are far beyond what you have seen. Systems can have the capability of creating complex three-dimensional programs for three- to five-axes CNC machining. Three-dimensional programming is common in both the toolmaking (forming die and mold) and aircraft manufacturing industries. In these industries the three-dimensional shapes must be cut from solid blocks of material.

Figure 10.65 shows an example of a five-axes machining problem where a tapered boss intersects a bullet nose shaped surface. The programmers objective is to machine the side of the boss and the intersection of the boss and bullet nose at the same time. To do this the cutter axis must be tilted while moving the machine in the X, Y, and Z axes. Many machine tool statements are required to perform this operation but only a few computer statements need to be created. Now you can see the power and importance of a CAD/CAM system's geometry and tool path creation capability to industry. We are now able to make things never thought possible a few years ago.



(Courtesy of Computervision Corp., Woburn, MA, ©1982. All rights reserved.)

Figure 10.65 Example of five-axes tool path display on three-dimensional operation.

QUESTIONS

- 1 List the advantages of computer aided part programming when compared with manual part programming.
- 2 List and briefly describe the main stages in the CAPP process.
- 3 Make a block diagram to illustrate a basic 'stand-alone' CAPP system.
- 4 Explain what is meant by 'time-sharing' and list the advantages and disadvantages of using such a facility.
- 5 Describe three methods that may be used for menu selection from a digitizing tablet.
- 6 Describe three methods of cursor control that may be used to identify graphic features on a CRT screen.
- 7 Describe three general methods for geometrically defining a point, a straight line and a circle or curve.
- 8 List the various data that would be included as tool change data statements during the preparation of a part program.
- 9 Explain what is meant by "cutter location data."
- 10 Explain the function of "postprocessing" and state how a range of machines having a variety of control systems would be accommodated.
- 11 Describe two ways in which computer graphics are used as an aid to part programming.
- 12 Describe briefly the meaning of CAD/CAM.

11

ADVANCED TECHNIQUES

The power of a computer's calculation ability continues to grow at a constant rate. With the computer's growth comes added programming techniques for CNC equipment. This chapter will give an overview of some of the additional programming abilities that exist today. One needs to follow the computer industry news in order to keep abreast of new developments and stay current in the field of programming.

PARAMETRIC PROGRAMMING

A parameter is a quantity that is constant in one particular case but variable in others. A simple engineering example of a parameter is the length of a bolt. One version of the bolt will have a certain length; all other versions will be identical, that is, they will have the same thread form, diameter, and hexagon head, but they will all vary in length. Thus the length of the bolt is a parameter, constant in one particular case but variable in others.

Parametric programming involves defining parameters and then using those parameters as the basis for one part program that may be used to machine not only the original component but a number of variations as well.

Figure 11.1(a) shows a component the dimensional features of which have been defined as parameters using the symbol # and a number: #1, #2, #3, and so on.

Figures 11.1(b)–11.1(g) show six variations of the component, the variations being indicated. A range of components such as this is referred to as a family of parts.

The machine movements necessary to machine each of the variations are all included in the original component. Some components require exactly the same movements, but with varying lengths of travel. Other components do not require all of the movements to be made. Using the more usual programming techniques, the production of each component would require a separate part program. Using the parametric part programming technique, instead of defining each dimensional movement individually in the X and Z axes, the parametric reference is programmed. Thus, to turn along the stepped diameter, the entry in the main program, referred to as the "macro," would read as follows:

```
N07 G01 X #4 Z #2
```

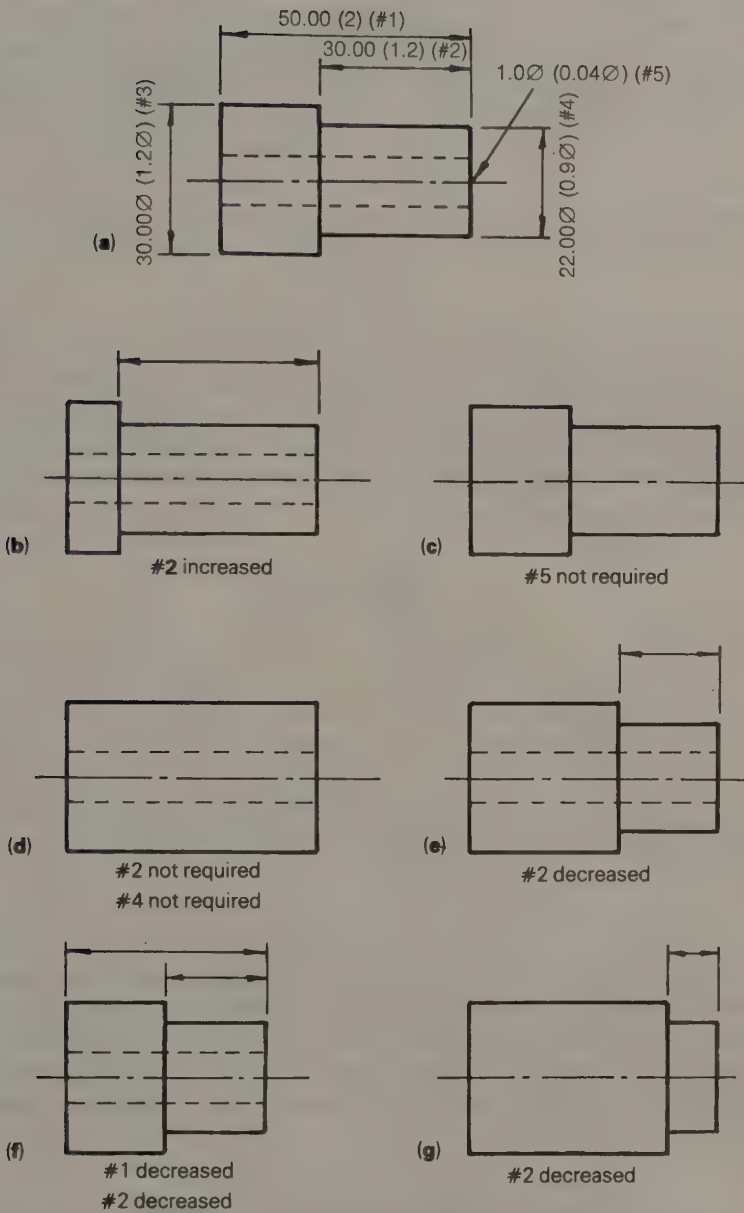


Figure 11.1 A "family of parts." (Inch units are given in parentheses.)

This entry would suffice for all components requiring a stepped diameter. Equally, one entry using parametric identification would suffice for facing all the components to length or drilling the hole.

Having programmed all movements and the sequence in which they are to occur, it remains to dimensionally define them. The dimensional details are entered as a list at the start of the part program. Thus the parameters and their dimensional values for the original components would read as follows (in metric units):

```
#1 = -50.00
#2 = -30.00
#3 = 30.00
#4 = 22.00
#5 = 10.00
```

As each parameter is called in the program, the dimensional entry made previously will be invoked.

To machine any of the variations in the family of parts requires a simple amendment of the original parametric values. The parameters (in metric units) to machine the component shown in Figure 11.1(b) would be:

```
#1 = -50.00
#2 = -40.00 (amended)
#3 = 30.00
#4 = 22.00
#5 = 10.00
```

and to machine the component in Figure 11.1(f):

```
#1 = -40.00 (amended)
#1 = -20.00 (amended)
#3 = 30.00
#4 = 22.00
#5 = 10.00
```

Now consider the components where the programmed movements necessary for machining the basic component are not required. By using a relatively simple programming technique, the control unit can be caused to skip the redundant blocks. The necessary program entry involves the use of certain conditional expressions in which assigned abbreviations are used, such as the following:

```
EQ = equal to
NE = not equal to
GT = greater than
LT = less than
GE = greater than or equal to
LE = less than or equal to
```


Consider Figure 11.1(d) and assume the #1 and #3 have been machined. In the program the next call will be to machine the stepped diameter. To avoid this, blocks must be skipped so an entry in the program will read as follows:

N15 IF [#4 EQ 0] GO TO N18

This statement says that if #4 is zero, move on to block number 18. Since #4 is nonexistent in the component, the parametric value will be entered as zero and, consequently, the control unit will move ahead.

The preceding description of the use of the parametric programming technique is a very simple one. It is in fact a very powerful concept and its full application is quite complex. For instance, parameters may be mathematically related, that is, they may be added together, subtracted from one another, and so on.

In addition, the parametric principle may be extended to include speeds and feeds, when all the likely variations for roughing, finishing, etc. may be given a parametric identity and called into the program as and when required.

Parametric-type programming although not uncommon is not standardized in its methods between machines or programming systems, so vendor manuals should be consulted.

DIGITIZING

Digitizing is the name given to a technique used to obtain numerical data direct from a drawing or model. To obtain numerical data from a drawing, which may or may not be dimensioned, it is placed on a special tablet, or table, and a probe is traced over the drawing outline. This movement is received by a computer and is transformed into digital or dimensional values. Only two-dimensional data can be obtained from a drawing. For three-dimensional data a model of the component is required, and a probe, which is electronic in operation, is traced over the surface of the model, this movement being recorded by the computer as before.

Numerical data obtained by digitizing can be used as the basis of a numerical control program. The numerical data entered into a computer, on the other hand, can be used to create the geometric data for a three-dimensional data base. This data base can then be used to create three-dimensional contouring cutter paths, using CAM/surfacing programming. The technique is only suitable for certain types of machining, such as profile milling, but the concept is likely to be developed to cater to a wider range of machine-shop activity.

FLEXIBLE MANUFACTURING SYSTEM

A flexible manufacturing system (FMS) is a computer-controlled machining arrangement which will cater to a variety of continuous metal-cutting opera-

tions on a range of components without manual intervention. The objective of such a system is to produce components at the lowest possible cost, and in particular components that are required only in small quantities. Thus a prime requirement of such a system is flexibility, that is, the capacity to switch from one type of component to another, or from one type of machining to another, without interruption in the production process.

Production costs per unit item decrease as the number of components required increases. Large production runs justify extensive capital expenditure on special-purpose machinery that does a particular job very efficiently and quickly. Machines of this type, however, are rarely adaptable to other types of work: they lack flexibility. When flexibility does exist—one skilled worker and one machine, for instance, where single components can be handled in random order—the production rate is slow and therefore costly. Modern flexible manufacturing systems aim to bridge the gap between these two extremes.

Flexible manufacturing systems have been made possible by the fact that modern machine control units can store in the computer memory a number of part programs which can be activated via a master computer program in random order, a system referred to as direct numerical control (DNC). The same master computer is also able to control the supply of workpieces to the machine. The third important factor, tooling, will be controlled by the part program itself, but if a wide range of machining is to be carried out the tooling magazine will be required to accommodate a large number of tools; for milling operations at least 60 and perhaps more than 100 may be necessary. To solve the problems of tool search time and maintenance of large tool conveyors some machine tool builders have made the conveyors themselves changeable.

A flexible manufacturing system will include at least two machines. When just two or three machines are involved, the arrangement is sometimes referred to as a "machining cell." A fully integrated system will include more machines than this and they will vary in type. Figure 11.2 illustrates the principle. Installations of this nature are, of course, very costly, but are becoming commonly used, at present. However, the modular approach to building such a system, that is, starting with two machines and then adding additional machines as and when investment funds are available, would suggest that the concept is set to become a dominant feature of machine-shop engineering.

The automatic supply of work to each machine is an essential feature of any system, large or small. The use of pallets is the most favored method, particularly for machining centers as opposed to turning centers, although they are also used for turning work. When pallets are used for turning work, the final loading of the machine usually involves a robot. Figure 11.3 shows how one robot may be positioned to service two machines.

The way pallets are used for milling operations will vary according to the type of work being handled and the space available. Figure 11.4 illustrates the use of a rack or storage retrieval system. Such a system as this is relatively simple and capable of modification and extension as and when required. More

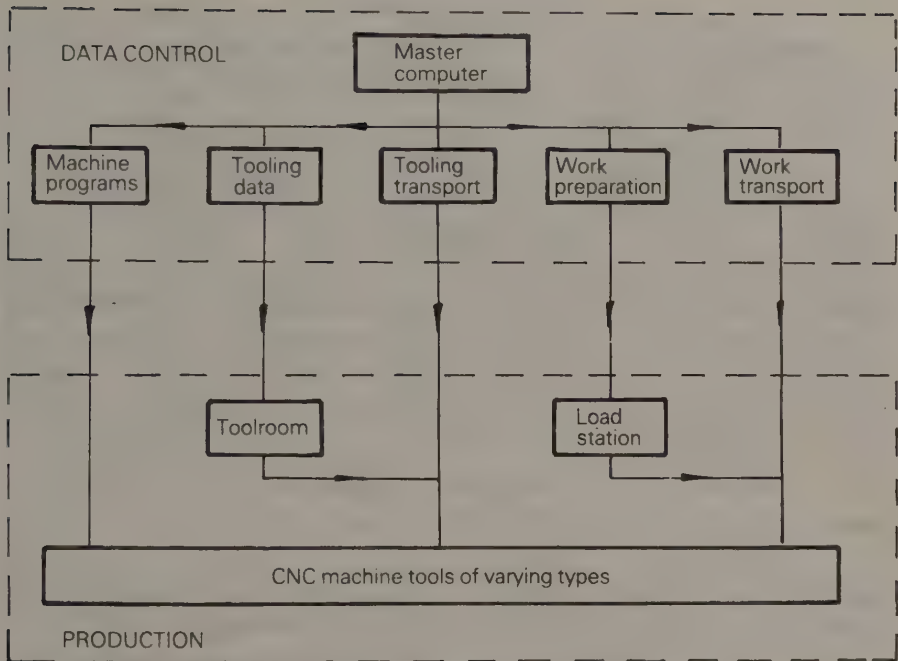


Figure 11.2 Computerized control of a manufacturing system.

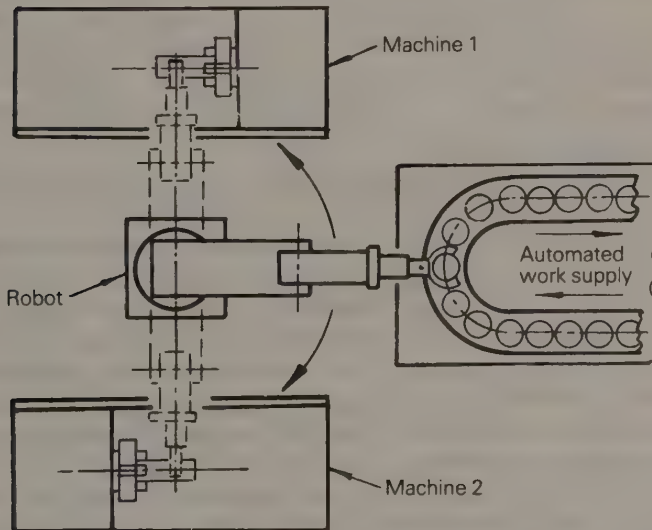


Figure 11.3 Robot loading of turning centers.

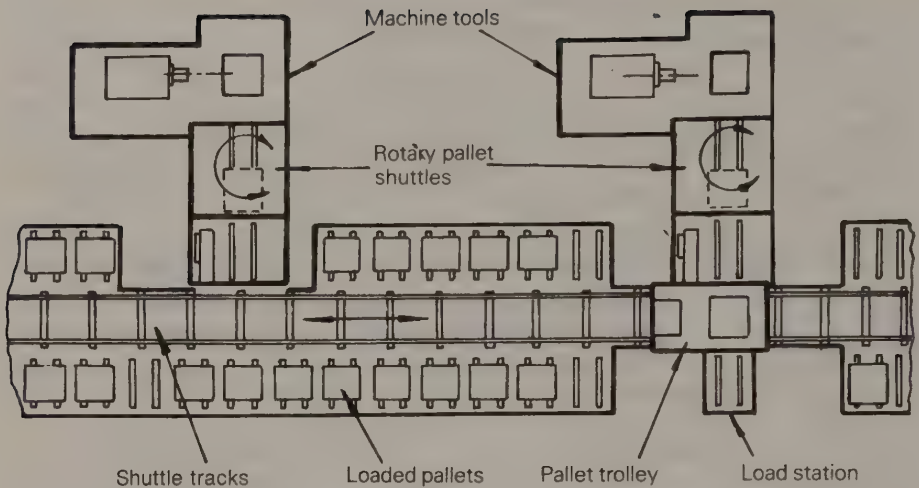


Figure 11.4 Flexible manufacturing system using racked palletized work supply.

complex systems involve the use of pallet conveyors and automatic guided vehicles (AGV) moving about the factory along predetermined routes, guided by inductive control wires buried in the workshop floor or computerized programmed controls. Each machine will have its own load/unload station and there will be a master load/unload station to which each pallet or AGV returns at the end of a journey. Figure 11.5 illustrates the principle.

Pallets may be fed to the machines in a predetermined order which means their positioning in the work queue is critical, otherwise a workpiece may be subjected to the wrong machining cycle! More commonly today, though, they are fed to the machine in random order and identified, usually by a photoelectric device responding to a bar code number attached to the pallet, when they arrive at the machine. On being identified the correct machining program for that workpiece will be called.

The preparation of work pallets is generally a manual operation. Their positioning and clamping in the machine are totally automated and, as a result, monitoring of their installation is necessary before machining commences. Limit switches or proximity sensors are a common feature of such control, and one method involving a combination of pneumatic and electrical principles is illustrated in Figure 11.6. Photoelectric cells, mechanical limit switches, and tactical sensors are also commonly used to check pallet and part locations.

Mention was made earlier of the high cost of flexible manufacturing systems which has, to date, limited their introduction. An intermediate approach to total automation is the use of automatic pallet-loaders dedicated to one machine. The pallet pool may involve two or more pallets and be designed in parallel rail, rectangular, oval or round form and when fully loaded will provide for an extended unsupervised production run lasting several hours, perhaps over-

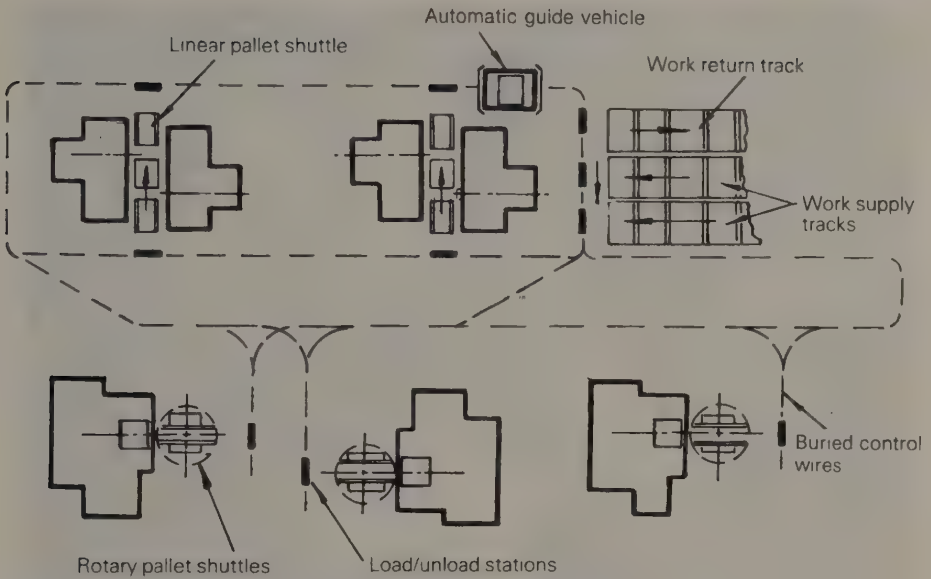


Figure 11.5 Flexible manufacturing system using remote controlled AGV work supply.

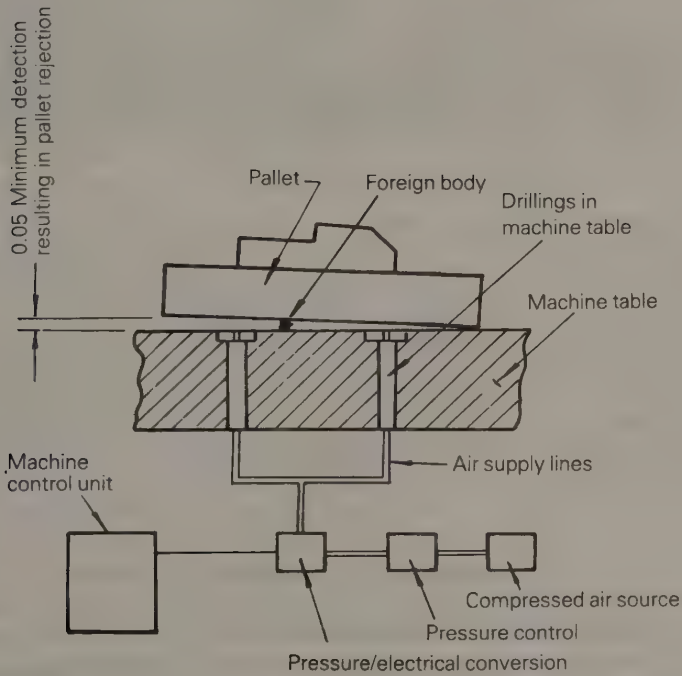


Figure 11.6 Pallet location control.

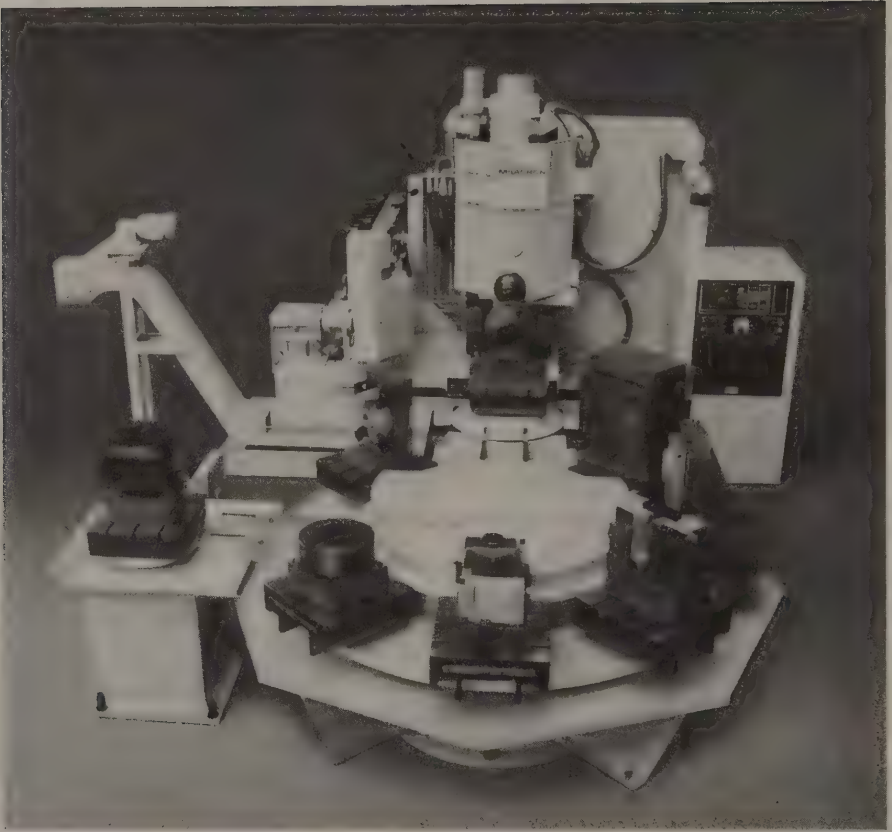


Figure 11.7 *Processing center providing unmanned production runs.*

night or throughout a weekend. Figure 11.7 illustrates such an arrangement, referred to as a “processing center” by the manufacturers; its flexibility is indicated by the range of components shown in position on the pallets. Figure 11.8 shows a robot-loaded turning center where work chutes provide a similar capability. Due to the costly nature of automation/machining cells computer software has been developed to draw up a complete animated system for testing. This software is called simulation software and allows for completely analyzing the system for bottlenecks and problems before the system is engineered and built.

ADAPTIVE CONTROL

Adaptive control is the term used to describe the facility that enables a machine control unit to recognize certain variations from the original conditions which may occur during a machining process and to make a compensating response. Unless such a response is made, the effect of the variations may be to damage

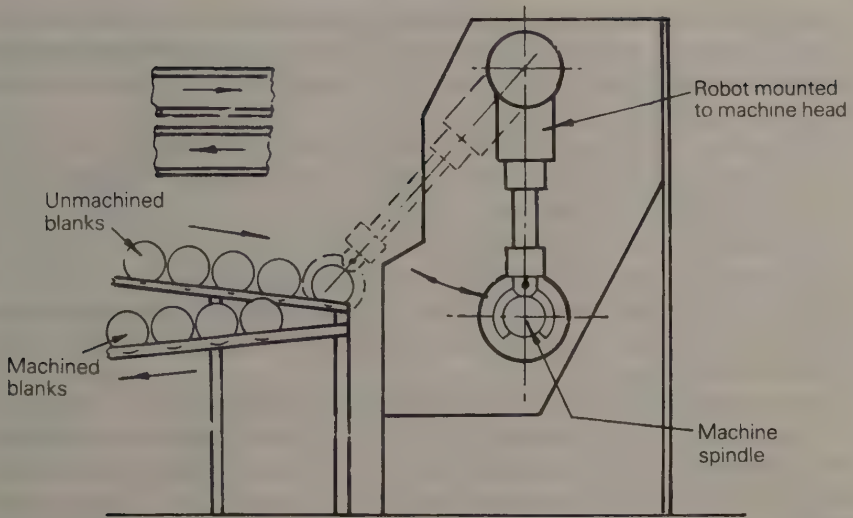


Figure 11.8 Robot loading facility providing unmanned production runs on a turning center.

the machine, tooling, workpiece, or cause the part to go out of prescribed tolerance. Adaptive control is basically a data “feedback system” rather like the closed-loop facility described in Chapter 1, which monitored slide positioning.

A number of unacceptable things can occur during a metal-cutting operation. For example, a tool may lose its cutting edge. On manually operated machines this would immediately be obvious to the operator, who would react accordingly. It is this type of response that adaptive control endeavours to emulate.

Now supposing the tool becomes dull on an automatic machining process. What is the likely effect? At least three things are likely to occur. First, the power necessary to turn the machine spindle is likely to increase, that is, there would be a torque variation. Second, there is likely to be a build-up in temperature between the cutting tool and the workpiece, as the tool tends to rub rather than cut. Finally, the tool itself is likely to deflect. By monitoring, that is, measuring, these variations and taking corrective action damage can be averted, and good parts consistently produced.

Torque monitoring of spindle and servo motors is one method of adaptive control that is used. The power consumption is monitored electronically and the application of the technology involves programming the control unit with data that will define the maximum and minimum torque values permitted for any particular operation. Assume that during a metal-cutting sequence the maximum torque value at the cutting tool is reached, indicating perhaps that the tool is dull or the component material is harder than anticipated. The control unit will respond to the feedback signal by lowering the feedrate and/or varying the spindle speed.

Consider another situation where, after modifications to feed and spindle

speed, the torque continues to increase to a point where the spindle is overloaded. In this case the control unit would inhibit the sequence and indicate a "warning" signal on the CRT screen. The problem can then be investigated and the conditions rectified.

The torque monitoring feature can also be used to detect the minimum torque that is programmed to occur after a certain length of slide travel. If the programmed torque does not occur, there may be two possible reasons. One is that the cutting tool is broken and has not made contact, a broken drill for example; the other is that the workpiece itself is not in position. Total inhibition of the machining process may be the necessary response, or alternatively a duplicate tool, referred to as a "sister" tool, already in the magazine or turret could be called.

Torque monitoring taken to its extreme means that spindle speeds and feeds can be omitted from part programs. Provided the control unit is programmed with the values of the maximum permitted speeds and feedrates, it is possible for the adaptive control to adjust them to suit the prevailing cutting conditions. For instance, when the torque is high, the feed would reduce, but when it is low, as for example when no cutting is taking place, the feed would be rapid.

Another approach to adaptive control is one which concerns itself with monitoring the presence of workpieces by the use of surface-sensing probes. Such a probe is illustrated in Figure 11.9 and is used in milling operations. The probe is mounted in the tool magazine alongside the cutting tools and can be called into operation via the part program, in the same way as a cutting tool, and be mounted in the machine spindle. The probe is electronic in operation and the stylus is interchangeable to accommodate different applications. It can be pro-



Figure 11.9 *Precision surface sensing probe.*

grammed to detect the presence, by touching on, of a surface in three axes and, if the surface is not present, it will inhibit the machining cycle.

Probes can also be programmed to check stock size and automatically cause the work datums to be offset to locate the finished part within the bounds of the stock material, thus ensuring that the final work surface is completely machined. If insufficient stock is present, the machining is not performed. This facility is particularly useful when machining castings or forgings.

Another application of a probe is to speed up a cycle by preventing non-metal-cutting passes of the cutting tool. For example, when machining casting or forgings, the part program will need to cater for all possibilities and may well include passing cuts that will be necessary only when there is excessive stock to be removed. If the probe detects that no metal is present, the feedrate will automatically be maximized or a program block may be skipped.

An interesting method of detecting the presence of cutting tools is a device that combines pneumatic or light, and electronic principles. It is designed primarily for use on machining centers to monitor small-diameter drills, taps, and reamers, which are very prone to break. The device is in the form of a simple caliper, which is positioned at a convenient point on the machine bed. The location of the caliper is predetermined and after use the cutting tool is moved via the part program so that it positions within the caliper. When in position, a jet of air is blown or light beam is shown from one side of the caliper and if the cutting tool is missing this jet of air or light beam will blow on to a pressure-sensitive/light-sensitive electrical device housed in the opposite arm of the caliper. This will generate a signal to the machine control unit that will result in either the machining process being halted or the tool being replaced in the program by a sister tool already housed in the magazine. If the tool is present, then there will be no air flow between the two arms of the caliper and the tool will automatically be replaced in the tool magazine to await a further call. Figure 11.10 illustrates the technique.

Another method of detecting broken or dull tooling involves the use of sound sensors. A cutting tool that is cutting properly will emit a certain sound. If the tool loses its edge or breaks, the sound it makes as it attempts to cut metal will be different from the original. The sound sensors detect the variation and will cause the program to be stopped or, alternatively, will call in a sister tool to replace the original.

Adaptive control is an area of computerized numerically controlled machining that is the subject of much research and experiment, and it is an area in which there are likely to be further very interesting developments.

IN-PROCESS MEASUREMENT

In-process measurement is the term used to describe the automatic measurement or gauging of a component while it is in position on the machine, and

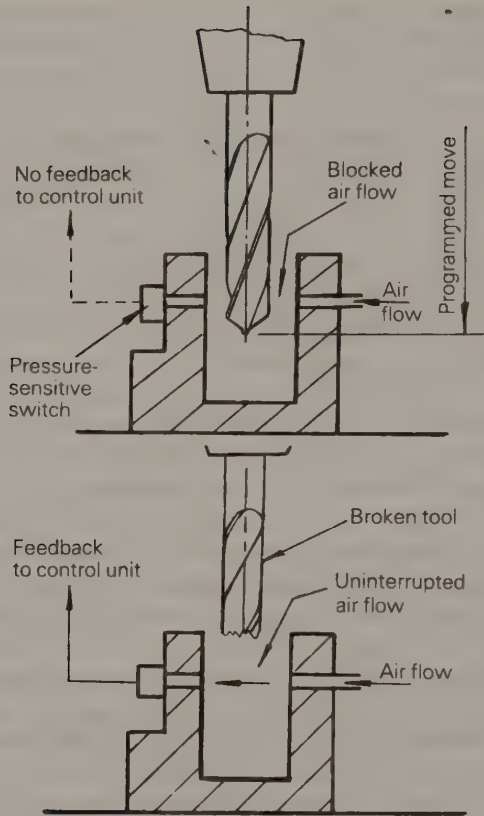


Figure 11.10 Broken tool detection unit.

while the correction of errors is still possible. It is not a new concept. The need for automatic measurement went hand in hand with the development of automatic machining processes, and fully automated machines performing a variety of operations were around long before the advent of computerized numerical control.

In-process measurement presents many difficulties. A machining area, with its accumulation of chips and coolant, is not an ideal place to carry out precision measurement involving delicate instruments or monitoring devices. Nevertheless, a number of very successful devices have been developed over the years, their method of operation being based on mechanical, pneumatic, optical, laser, and electronic principles.

Many of the earlier devices, though not all, were "open loop," that is, there was no feedback of data to the machine controls and so there was no automatic adjustment of the machine setting to compensate for unacceptable size variations. Correction was possible only by manual intervention, but at least this was usually possible without halting the machining process.

On modern CNC machines the accuracy with which slide movements are generally made and monitored can, in the case of some classes of work, eliminate the need for further control, since the slide movements, and therefore the relative tool movements, are made to an accuracy which may well be within the dimensional tolerances of the component. In other cases this degree of control is insufficient and, as was stated in Chapter 2, transducers which monitor slide movement or leadscrew rotation may not give an accurate indication of the tool and work relationship. For instance, a tool may wear, thus affecting the dimensional size of the component, but this will have no connection with slide movement and no compensation will be made. Similarly, the workpiece may not be precisely located, or may be impossible to locate precisely, so again some monitoring and correction of movement may be necessary to ensure that surfaces are relatively positioned. These are the sorts of situations that in-process measuring can monitor.

The modern in-process measuring device is electronic in operation. It consists of a probe, laser or vision, and is capable of monitoring positional variations in three axes. The way in which it is applied will depend on the machine type, but it can be applied to the measurement of internal and external diameters, lengths, depths of slots, hole centers, and so on. Programmed air blasts are used to clean the work prior to measurement.

One method of using touch sensors requires reference to be made to an established datum, which may be a surface which is part of the machine structure, for example a tailstock barrel, or a surface on the component. The program will bring the sensor into contact with the reference face and record its position as zero. It will then be moved to the surface being checked and the resulting move will be compared by the machine control unit with a pre-programmed value. If there is a variation, a compensation in the relative tool offset will be made.

Features such as a bored hole diameter can be measured by touching it on each side of the hole, and the resulting movement, plus the stylus diameter of the probe, will indicate the hole size. The necessary calculation will be made by the control unit and again a comparison will be made with a pre-programmed value and tool offsets initiated as required. The technique is illustrated in Figure 11.11.

Measurement of this nature is not completely divorced from the machine slide movement, and its accuracy can never be better than the resolution, that is, the smallest increment that can be determined by the control unit of the machine.

COMPUTER INTEGRATED MANUFACTURING (CIM)

It has been shown that computers are all around us in the CNC manufacturing area: Design "CAD," engineering analysis, CNC programming "CAM," sim-

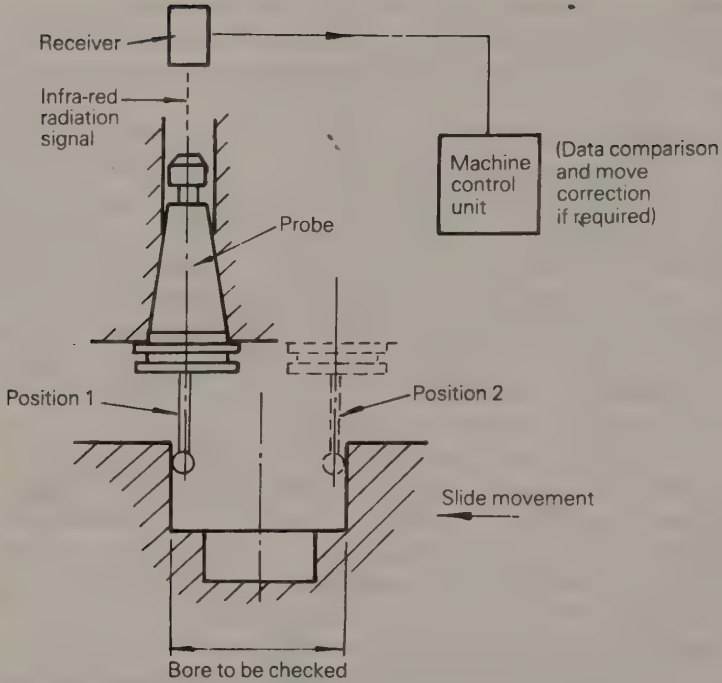


Figure 11.11 *In-process measurement by electronic probe.*

ulation of machining and manufacturing cells, and adaptive controls. We also know that computers are used in the area of business and management for accounting, order processing, forecasting, sales analysis, production schedule planning, inventory management, materials requirements planning, machine capacity requirements planning, production control and costing, purchasing, production monitoring, payroll, financial analysis, general ledger, and accounts payable.

Computer-integrated manufacturing (CIM) is the concept of a totally automated factory in which all business, engineering, and manufacturing processes are linked and controlled by a computer system. See Figure 11.12 for a graphic representation. CIM enables managers, production planners and schedulers, shop floor foremen, and accountants to all use the same database as production designers and engineers. Owing to the short lead times and complexity of manufacturing, all areas of a business must work together to make a profit.

The enterprise model of a computer system, which allows business, engineering, and manufacturing to share information and work together, is shown in Figure 11.13. The inner ring depicts the computer hardware, software, and database. The middle ring indicates the decision making level of management and computer support staff (report generation). The outer ring indicates the users and suppliers of information that must work together. All areas of busi-

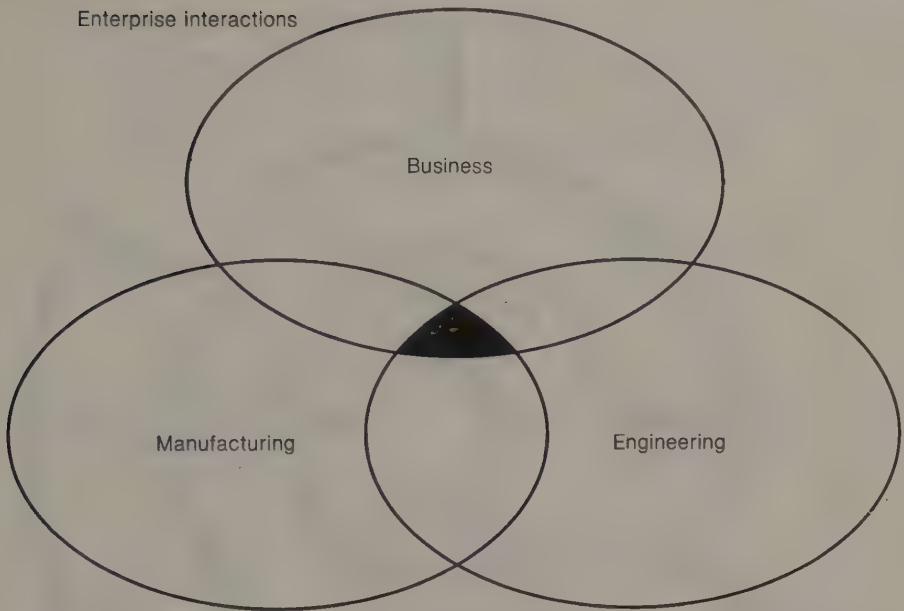


Figure 11.12 *The three areas of a corporation that must work together and share data in order to make a profit*

ness are indicated in the three rings, and information can flow between any of the areas. As in any business venture there should be improvements or goals to achieve. In implementing a CIM plan one would be looking for gains in the following areas:

- A reduction in product lead time.
- Faster, more reliable availability of quality information.
- The ability of process monitoring and tracking.
- Improved product delivery information and quality.
- Improved supplier performance and the ability to track historical performance and make predictions.
- Faster information flow and simultaneous engineering to reduce product cost.
- Improved communication of business plans to employees creating improved attitudes and quality consciousness.

The idea of CIM revolves around the center ring of shared information and people working in teams, simultaneously. The old method of passing information from department to department upon task completion is outdated and slow. To be responsive manufacturers we must keep from duplicating efforts and help each other for the common good of supplying products in a short

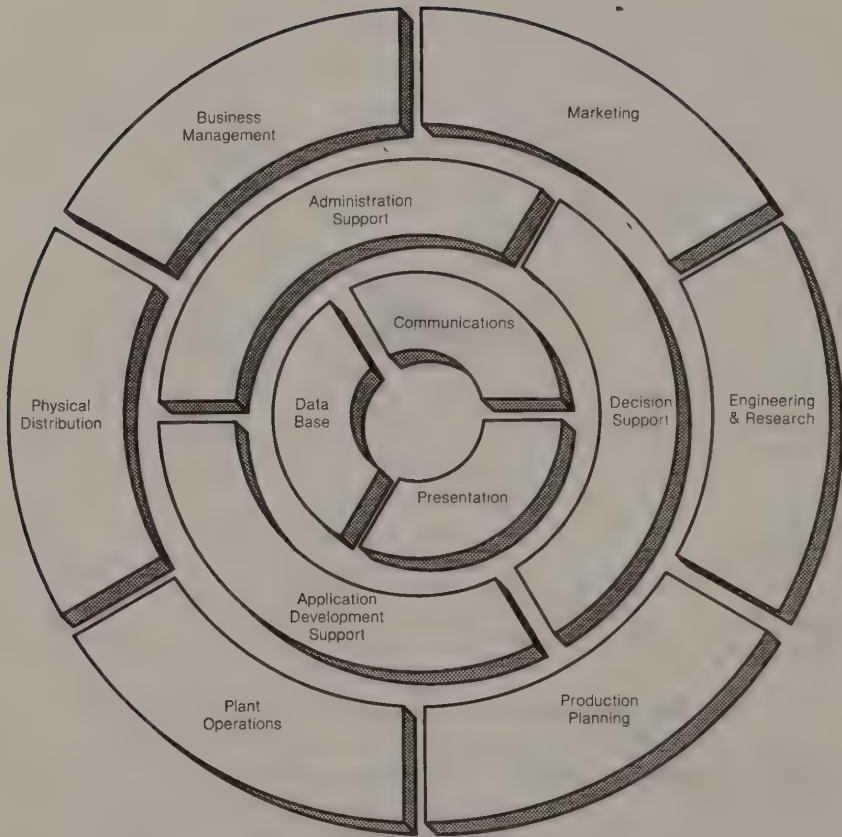


Figure 11.13 *The three rings of computer information. (Courtesy of IBM.)*

period of time. Figure 11.14 shows in simple terms the interaction of information within a business.

In Chapter 10 we discussed the use of a CAD/CAM system for part design and CNC programming. The tasks involved in product design, tool design, and manufacturing engineering are much more involved than that. A product must be conceived first but then it must be analyzed for failure and determined how best to make it. It is here that the product, tool, and manufacturing engineers need to work together and share information to solve all the problems to get a product manufactured. At the same time marketing and sales need to be involved in order to see if the product will sell. As sales are projected, we then need to make plans and order materials to build the product. When orders come in, the product needs to be tracked through its processes so, as delays occur and changes must be made, the right people in engineering, business, and manufacturing are informed. As the end product leaves the manufacturing arena and is distributed to customers, billing must occur. During this whole time

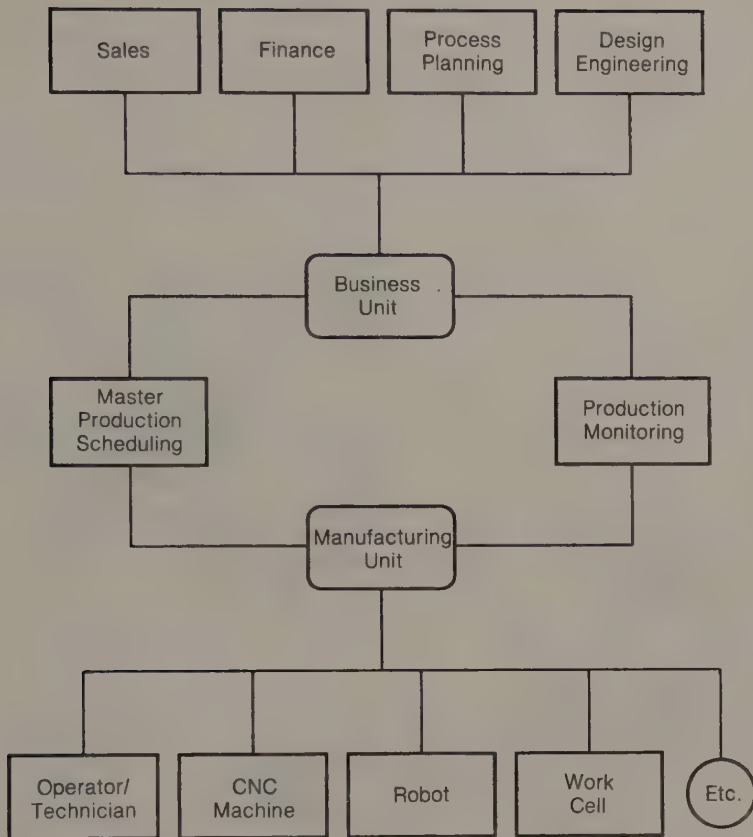


Figure 11.14 A CIM view: the business, engineering, and manufacturing units.

accounting and payroll continue. Now you can see the advantage of maintaining one shared computer database for all. Figure 11.15 shows the entire workflow of an enterprise, and it is easy to see how complex the information flow would be using paper instead of computers.

Looking further into business planning (Figure 11.16), we see three levels of management planning that the computer can help with. The top or first level of management deals with the future business plans, sales forecasting, and setting of future production levels. The second level is operations management, which takes care of monthly master schedules for production, makes plans for material needs, and checks capacity requirements (production equipment availability) to the master production schedule. The third level of management is operations execution at the production floor. This final level schedules the production floor day to day in order to supply the proper product at the right time. This person is in charge of initial quality and wants to know about daily deliveries of vendor parts. A common computer database for all these manage-

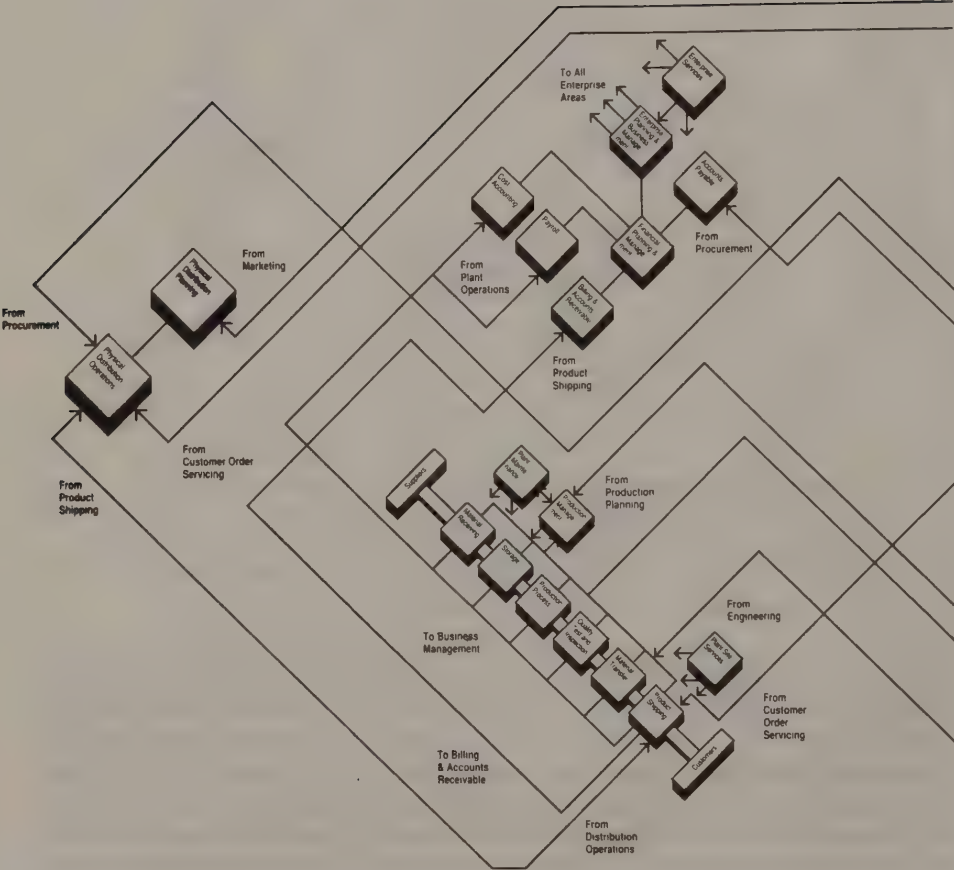
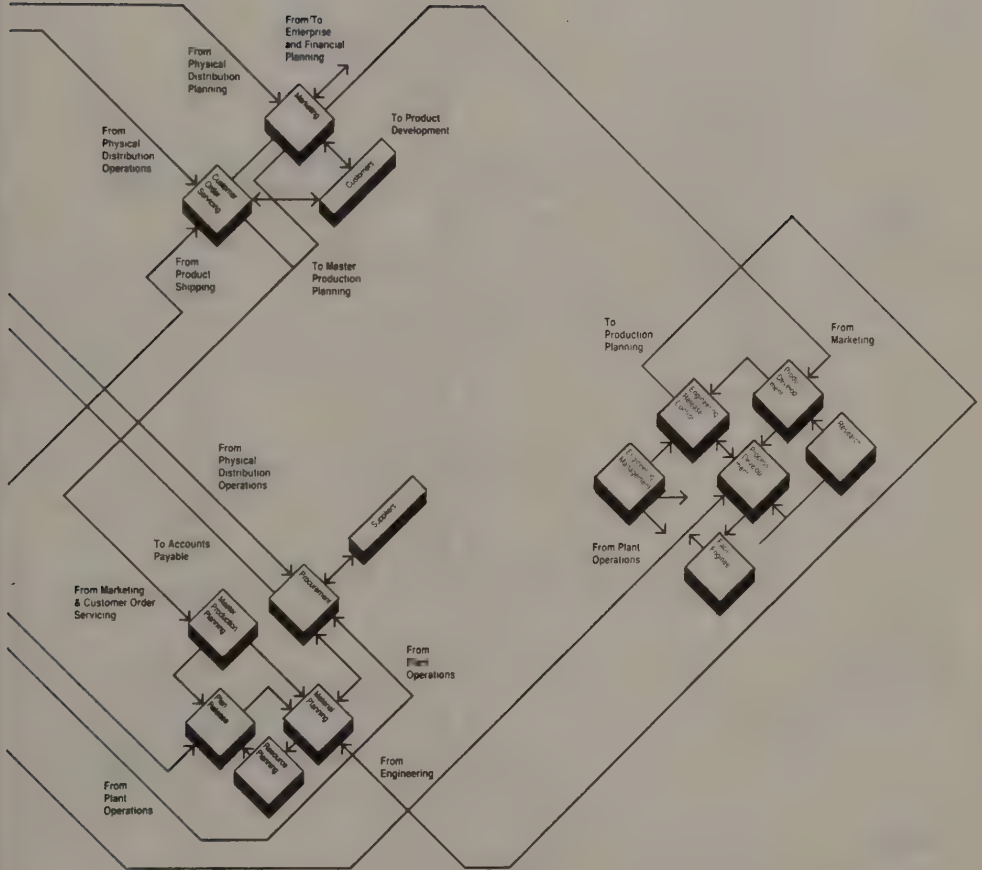


Figure 11.15 The Enterprise workflow. (Courtesy of IBM.)



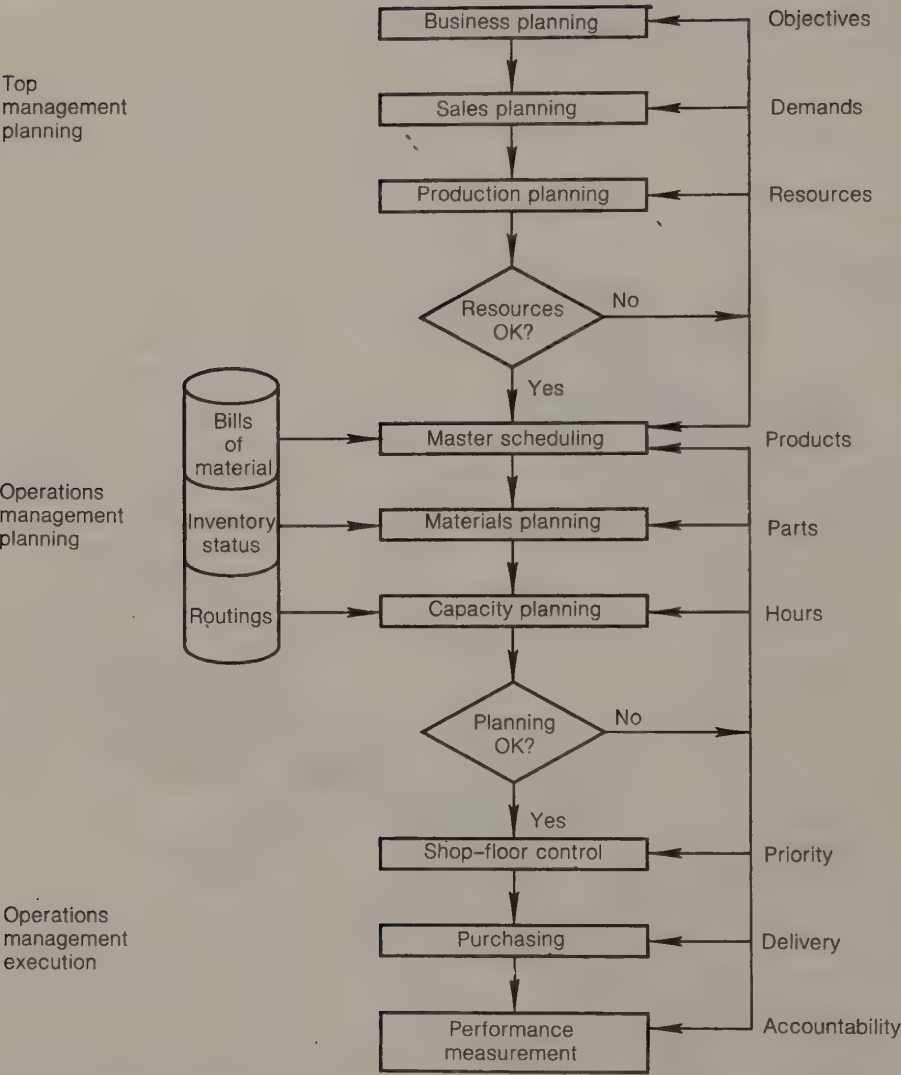


Figure 11.16 Business management levels.

ment levels makes sure everyone has the same information for planning and problem solving.

In Figure 11.17 are the modules of a computer software package developed by IBM to take care of the previously mentioned business tasks in a company's development of CIM. The package diagrammed is called MAPICS (Manufacturing Accounting and Production Information Control System) and is made up of 16 modules, which can be broken down into four major areas. The dia-

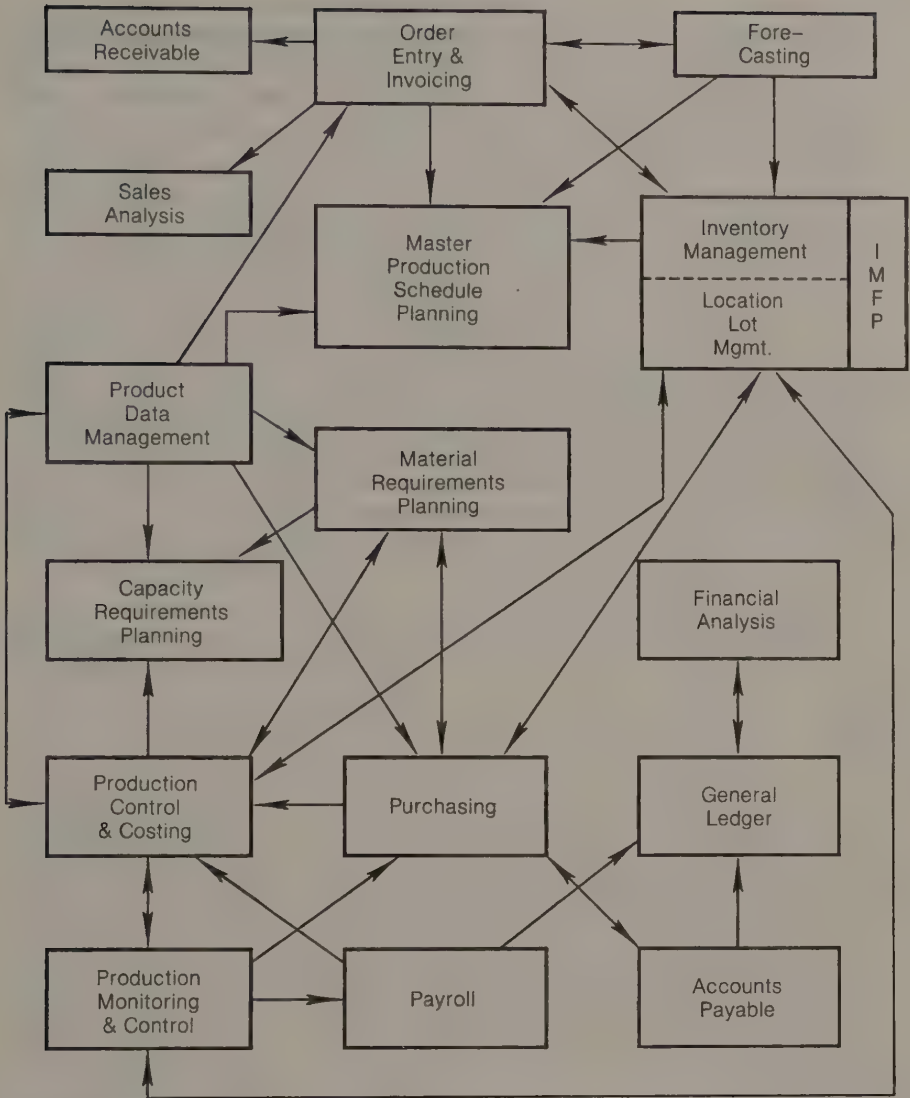


Figure 11.17 The MAPICS applications (showing major interfaces only).

gram shows you how various types of information will cross applications in the database use.

The first application is Plant Operations, which is made up of inventory management, production control and costing, production monitoring and control, and purchasing. Inventory management improves plant productivity through reduced and improved accuracy of parts inventory. Production Control and Costing allows management to track production orders. The computer will

highlight excessive material and labor costs for correction, pinpoint part locations, show production time remaining, and report daily quantities completed. This module will allow daily priority reports to be created so work can be sequenced for meeting delivery requirements. Production Monitoring and Control through timely and accurate shop reporting helps ensure work is progressing, and orders are met promptly. This application receives orders that can be added to, modified, or split, and then print shop instruction packets. The Purchasing section allows for maintaining valid quotes giving buyers more time to negotiate, and analyzes vendor performance. The system also allows for the tracking of purchase orders and validating of vendor invoices.

The second MAPICS area of application is Marketing and Physical Distribution, which is made up of the Order Entry and Invoicing, Sales Analysis, and Forecasting modules. The order entry point is the key starting point of a manufacturing organization. Detailed reports can improve cash flow, help manage inventory costs, and analyze product performance. The order entry updates both inventory and accounts receivable records. From sales records analysis reports can be generated to forecast/predict future sales and needs. This software will help ensure reasonable ship dates, coordinate inventory and production requirements, and automatically price customer orders.

The third MAPICS application is Production Planning made up of Master Production Schedule Planning, Material Requirements Planning, Capacity Requirements Planning, and Product Data Management. Production planning supplies information for optimizing resources and minimizing costs, so production stays on schedule and meets forecasts. Master production schedule planning allows production plans to be analyzed according to inventory levels, sales projections, and resource projections so that a master production schedule can be generated. Material requirements planning software creates a purchasing or manufacturing order recommendation report to be able to meet the master production schedule. Capacity requirements planning is used to calculate the future demands on machinery and manpower, both long and short range, to help reduce bottlenecks. Product data management is used to maintain the information database for engineering, manufacturing, and accounting. This file maintains information on all part numbers, bill of materials, work stations, and routings. With the product data information, cost simulations and analyses can be performed.

The last MAPICS application is Financial Management and Business Control made up of General Ledger, Financial Analysis, Accounts Payable, Payroll, and Accounts Receivable. It is here that up to date financial information is maintained for management decision making. This financial database gives tighter controls while making sure everyone gets the same fast, accurate information. General ledger allows you to get clear reports on the company's operating performance. Financial analysis will report on significant trends so additional production planning can occur. Accounts payable gives accurate, timely information on invoice due dates, vendors, and amounts. Accounts payable software

is a flexible way to manage cash outflow and analyze vendor performance. The payroll unit performs all payroll tasks as well as attendance reporting and payroll cost monitoring. Accounts receivable helps minimize collection periods, monitor cash flow in, and maintain records of good and bad customers.

It is easily seen that it would be a major advantage to have one computer system and database to control all the business functions. Everyone has the same up to date information to do their own planning and report generation. This IBM product runs on a midrange business computer. It should be noted that other companies also supply similar forms of software. The size of the computer running the software will have an effect on its capabilities. Software for some or all of these abilities are available on computer systems from personal size to main frames.

Remember, it is when we combine business control systems, engineering computer systems (CAD/CAM and computer-aided engineering analysis) and automated plant floor systems that we actually have true CIM. Many companies start out with pieces of CIM and will be building a system over a number of years. It is the computer that puts speed and ease into integration but it will take human integration also for the process to work.

QUESTIONS

- 1 What is a flexible manufacturing system?
- 2 What characteristic of computer technology makes the concept of FMS feasible?
- 3 What do you understand by the term in-process measurement? Explain how it might be applied to checking the size of a turned diameter.
- 4 What is torque monitoring? State two instances where its application may be useful in an automated machining process.
- 5 How are sound sensors used to monitor cutting-tool condition?
- 6 What is a sister tool?
- 7 Describe the technique of digitizing.
- 8 Describe parametric programming and explain the advantages of the technique.
- 9 Write a brief description of CIM.
- 10 What three major areas of a company make up the three rings of CIM?
- 11 What is it that takes place in a computerized business control system such as MAPICS?

APPENDIX A

ELECTRONICS INDUSTRIES ASSOCIATION SPECIFICATION

EIA Standards RS-274-D, "Interchangeable Variable Block Data Format for Positioning, Contouring, and Contouring/Positioning Numerically Controlled Machines," and RS-358-B, "Sublet of American National Standard Code for Information Interchange for Numerical Machine Control Perforated Tape" are two standards that contain useful information pertinent to the subject matter of this book. Copies of these standards may be obtained from the Electronic Industries Association, 200 Eye Street, N.W., Washington, DC 20006.

APPENDIX B

CUTTING SPEED AND FEED INFORMATION, CARBIDE GRADES, AND POWER REQUIREMENTS FORMULAS

Information Sheet

How to Calculate Revolutions per Minute when the Cutting Speed Is Given (surface feet per minute)

From the past experience and tests, the following values of cutting speeds (CS) for the materials and operation shown in the tables will be used as *maximum values* in this shop. These values are given to use for *high-speed-steel tools*:

$$\text{RPM} = \frac{4 \text{ (CS)}}{D} \qquad \text{CS} = \frac{D(\text{RPM})}{4}$$

D = diameter of tool.

Material	Turning	Drilling	Shaping & Milling	Reaming
Mild Steel or Cold Rolled Steel	90 ft/min	80 ft/min	70 ft/min	40 ft/min
S. A. E. 1144	100 ft/min	90 ft/min	90 ft/min	50 ft/min
S. A. E. 4130	50 ft/min	45 ft/min	40 ft/min	25 ft/min
1% Carbon Steel	50 ft/min	45 ft/min	40 ft/min	25 ft/min
High Speed Steel	45 ft/min	40 ft/min	40 ft/min	20 ft/min
Rolled Brass	150–200 ft/min	120 ft/min	125 ft/min	70 ft/min
Cast Brass	100–125 ft/min	100 ft/min	100 ft/min	50 ft/min
Cast Iron	75 ft/min	70 ft/min	60 ft/min	35 ft/min
Cast Aluminum	250 ft/min	250 ft/min	200 ft/min	100 ft/min
Wrought Aluminum	400 ft/min	400 ft/min	400 ft/min	200 ft/min

Lathe cutting speeds for carbide tools (surface feet per minute)

Material Machined	Depth of Cut (in.)	Feed per Revolution (in.)	Surface Feet per Minute
Aluminum	.005-.015	.002-.005	700-1000
	.020-.090	.005-.015	450-700
	.100-.200	.015-.030	300-450
	.300-.700	.030-.090	100-200
Brass, bronze	.005-.015	.002-.005	700-800
	.020-.090	.005-.015	600-700
	.100-.200	.015-.030	500-600
	.300-.700	.030-.090	200-400
Cast iron (medium)	.005-.015	.002-.005	350-450
	.020-.090	.005-.015	250-350
	.100-.200	.015-.030	200-250
	.300-.700	.030-.090	75-150
Machine steel	.005-.015	.002-.005	700-1000
	.020-.090	.005-.015	550-700
	.100-.200	.015-.030	400-550
	.300-.700	.030-.090	150-300
Tool steel	.005-.015	.002-.005	500-750
	.020-.090	.005-.015	400-500
	.100-.200	.015-.030	300-400
	.300-.700	.030-.090	100-300
Stainless steel	.005-.015	.002-.005	375-500
	.020-.090	.005-.015	300-375
	.100-.200	.015-.030	250-300
	.300-.700	.030-.090	75-175
Titanium alloys	.005-.015	.002-.005	300-400
	.020-.090	.005-.015	200-300
	.100-.200	.015-.030	175-200
	.300-.700	.030-.090	50-125

Note: Cutting speeds obtained depend on operation setup and machine tool.

Suggested feed per tooth for *high speed* steel milling cutters

Material	Face Mills	Helical Mills	Slotting and Side Mills	End Mills	Form Relieved Cutters	Circular Saws
Plastic	.013	.010	.008	.007	.004	.003
Magnesium and Alloys	.022	.018	.013	.011	.007	.005
Aluminum and Alloys	.022	.018	.013	.011	.007	.005
Free Cutting Brasses and Bronzes	.022	.018	.013	.011	.007	.005
Medium Brasses and Bronzes	.014	.011	.008	.007	.004	.003
Hard Brasses and Bronzes	.009	.007	.006	.005	.003	.002
Copper	.012	.010	.007	.006	.004	.003
Cast Iron, Soft (150-180 B.H.)	.016	.013	.009	.008	.005	.004
Cast Iron, Medium (180-220 B.H.)	.013	.010	.007	.007	.004	.003
Cast Iron, Hard (220-300 B. H.)	.011	.008	.006	.006	.003	.003
Malleable Iron	.012	.010	.007	.006	.004	.003
Cast Steel	.012	.010	.007	.006	.004	.003
Low Carbon Steel, Free Machining	.012	.010	.007	.006	.004	.003
Low Carbon Steel	.010	.008	.006	.005	.003	.003
Medium Carbon Steel	.010	.008	.006	.005	.003	.003
Alloy Steel, Annealed (180-220 B. H.)	.008	.007	.005	.004	.003	.002
Alloy Steel, Tough (220-300 B. H.)	.006	.005	.004	.003	.002	.002
Alloy Steel, Hard (300-400 B. H.)	.004	.003	.003	.002	.002	.004
Stainless Steels, Free Machining	.010	.008	.006	.005	.003	.002
Strainless Steels	.006	.005	.004	.003	.002	.002
Monel Metals	.008	.007	.005	.004	.003	.002

Courtesy of Carboloy Inc., A Seco Tools Co.

Suggested feed per tooth—milling

Material	Face Mills		Helical mills		Slotting & side mills		End mills	Form relieved cutters	Circular saws	
	Hss	Carb	Hss	Carb	Hss	Carb	Hss	Carb	Hss	Carb
Plastics	.013	.015	.010	.012	.008	.009	.007	.004	.003	.004
Magnesium and Alloys	.022	.020	.018	.016	.013	.012	.011	.010	.006	.005
Aluminum and Alloys	.022	.020	.018	.016	.013	.012	.011	.010	.006	.005
Free Cutting Brasses & Bronzes	.022	.020	.018	.016	.013	.012	.011	.010	.006	.005
Medium Brasses and Bronzes	.014	.012	.011	.010	.008	.007	.006	.004	.003	.003
Hard Brasses and Bronzes	.009	.010	.007	.008	.006	.006	.005	.003	.002	.003
Copper	.012	.012	.010	.009	.007	.007	.006	.004	.003	.003
Cast Iron, Soft (150–180 B.H.)	.016	.020	.013	.016	.009	.012	.008	.010	.005	.004
Cast Iron, Medium (180–220 B.H.)	.013	.015	.010	.013	.007	.010	.007	.008	.004	.003
Cast Iron, Hard (220–300 B.H.)	.001	.012	.008	.010	.006	.007	.006	.003	.003	.003
Malleable Iron	.012	.014	.010	.011	.007	.008	.006	.004	.003	.004
Cast Steel	.012	.014	.010	.011	.007	.008	.006	.004	.003	.004
Low Carbon Steel, Free Machining	.012	.016	.010	.013	.007	.009	.006	.008	.005	.004
Low Carbon Steel	.010	.014	.008	.011	.006	.008	.005	.007	.003	.004
Medium Carbon Steel	.010	.014	.008	.011	.006	.008	.005	.007	.003	.004
Alloy Steel, Annealed (180–220 B.H.)	.008	.014	.007	.011	.005	.008	.004	.007	.003	.004
Alloy Steel, Tough (220–300 B.H.)	.006	.012	.005	.010	.004	.007	.003	.006	.002	.003
Alloy Steel, Hard (300–400 B.H.)	.004	.010	.003	.008	.003	.006	.002	.003	.001	.003
Stainless Steels, Free Machining	.010	.014	.008	.011	.006	.008	.005	.007	.003	.004
Stainless Steels	.006	.010	.005	.008	.004	.006	.003	.005	.002	.003
Monel Metals	.008	.010	.007	.008	.005	.006	.004	.005	.003	.003

Courtesy of Carbidey Inc., A Seco Tools Co.

Suggested feed per tooth for sintered carbide tipped cutters

Material	Face Mills	Helical Mills	Slotting and Side Mills	End Mills	Form Relieved Cutters	Circular Saws
Plastic	.015	.012	.009	.007	.005	.004
Magnesium and Alloys	.020	.016	.012	.010	.006	.005
Aluminum Alloys	.020	.016	.012	.010	.006	.005
Free Cutting Brasses and Bronzes	.020	.016	.012	.010	.006	.005
Medium Brasses and Bronzes	.012	.010	.007	.006	.004	.003
Hard Brasses and Bronzes	.010	.008	.006	.005	.003	.003
Copper	.012	.009	.007	.006	.004	.003
Cast Iron, Soft (150-180 B. H.)	.020	.016	.012	.010	.006	.005
Cast Iron, Medium (180-220 B. H.)	.016	.013	.010	.008	.005	.004
Cast Iron, Hard (220-300 B. H.)	.012	.010	.007	.006	.004	.003
Malleable Iron	.014	.011	.008	.007	.004	.004
Cast Steel	.014	.011	.008	.007	.005	.004
Low Carbon Steel, Free Machining	.016	.013	.009	.008	.005	.004
Low Carbon Steel	.014	.011	.008	.007	.004	.004
Medium Carbon Steel	.014	.011	.008	.007	.004	.004
Alloy Steel, Annealed (180-220 B. H.)	.014	.011	.008	.007	.004	.004
Alloy Steel, Tough (220-300 B. H.)	.012	.010	.007	.006	.004	.003
Alloy Steel, Hard (300-400 B. H.)	.010	.008	.006	.005	.003	.003
Stainless Steels, Free Machining	.014	.011	.008	.007	.004	.004
Stainless Steels	.010	.008	.006	.005	.003	.003
Monel Metals	.010	.008	.006	.005	.003	.003

Courtesy of Carbonyl Inc., A Seco Tools Co.

Cutting data for drilling (meters per minute)

(Reproduced by kind permission of Guhring Ltd.)

Material of workpiece	Drill type	Material of drill	Point angle	Cutting speed m per min.	Feed scale No. (see p. 468)	Coolant
Free-cutting mild steel hardness up to 500 N/mm ²	N/GT 50	HSS	118	30 – 50	4	Soluble oil
Non-alloyed carbon steel with $\leq 0.4\%$ carbon ≤ 800 N/mm ²	N	HSS	118	20 – 30	4	Soluble oil
Non-alloyed carbon steel with $> 0.4\%$ carbon, hardness 800–1000 N/mm ² and purified alloy steel with a hardness ≤ 700 N/mm ²	N/GT 100	HSS	118	16 – 20	3	Soluble oil
Non-alloyed tool steels with a hardness of 800–1000 N/mm ² and refined alloy steels with a hardness of 700–1000 N/mm ²	N/GT 100	HSS	118	12 – 16	3	Soluble oil
Alloyed tool steels hardness 800–1000 N/mm ² and refined alloy steels with a hardness of 1000–1200 N/mm ²	N/GV	HSCO (HSS)	118 (130)	10 – 16	2	Soluble oil
Refined alloy steels with a hardness of > 1200 N/mm ²	N/GV	HSCO	130	5 – 8	1	Soluble oil, cutting oil
Chrome-molybdenum, stainless steel	N	HSCO	130	8 – 12	1	Soluble oil, cutting oil
Stainless, austenitic, nickel-chrome, heat resisting steels	N/Ti (Specials)	HSCO	130	3 – 8	1	Cutting oil or cutting oil with molybdenum disulphide additives
Manganese steels containing up to 10% molybdenum	H (Specials)	HSCO	130	3 – 5	1	Dry preheat to 200–300
Spring steels	N/GV	HSCO (HSS)	130	5 – 10	1	Soluble oil, cutting oil
Nimonic alloys	W/Ti (Specials)	HSCO	130	3 – 8	1	Cutting oil or cutting oil with molybdenum disulphide additives
Ferro-tic	N/Ti	HSCO	118°/130	3 – 5	1	Dry: compressed air
Titanium and titanium alloys	Ti (Specials)	HSCO	130	3 – 5	1	Cutting oil or cutting oil with molybdenum disulphide additives
Grey cast iron up to GG 26 and malleable iron	N	HSS (HSCO)	118 /90 (double angle point)	16 – 25	5	Dry: soluble oil
Hard cast iron up to 350 brinell	N	HSCO	118 /90 (double angle point)	8 – 12	4	Dry soluble oil
Brass to MS 58	H	HSS	118	60 – 80	6	Dry, cutting oil
Brass from MS 60	H (N)	HSS	118	30 – 60	5	Soluble oil, cutting oil
Red copper	W/GT 50	HSS	130	30 – 60	5	Soluble oil, cutting oil

Cutting data for drilling (meters per minute) (contd.)

Material of workpiece	Drill type	Material of drill	Point angle	Cutting speed m per min.	Feed scale No	Coolant
Electrolytic copper	N	HSS	130°	20 – 30	5	Soluble oil, cutting oil
German silver	N	HSS	118°	20 – 30	3	Soluble oil, cutting oil
Copper nickel and copper-iron alloys	N	HSS	130	20 – 30	3	Soluble oil, cutting oil
Copper-aluminium alloys	N	HSS	130	10 – 30	3	Soluble oil, cutting oil
Alloys of copper and beryllium	H	HSS	130°	10 – 16	2	Soluble oil, cutting oil
Copper-manganese and copper-silicon alloys	N	HSS	130°	25 – 30	3	Soluble oil, cutting oil
Pure aluminium	W/GT 50	HSS	130°	40 – 60	5	Soluble oil
Aluminium-manganese and aluminium-chrome alloys	W/GT 50	HSS	130°	40 – 60	5	Soluble oil
Aluminium alloyed with lead, antimony or tin	W/GT 50	HSS	130°	60 – 100	5	Soluble oil
Aluminium-copper alloys containing silicon, magnesium, lead, tin, titanium or beryllium	W/GT 50	HSS	130°	40 – 60	5	Soluble oil
Aluminium-silicon alloys containing copper, magnesium, manganese or chrome	W/GT 50	HSS	130°	40 – 60	5	Soluble oil
Aluminium-magnesium alloys with silicon, manganese or chrome	W/GT 50	HSS	130°	60 – 100	5	Soluble oil
Magnesium and magnesium alloys (Electron)	W/GT 50	HSS	130°	80 – 100	5	Dry
Zinc, Zamac	N	HSS	118	30 – 40	4	Soluble oil
Hard duroplastics	H	HSS/HM	80	10-20 50-100	3/4	Dry: Compressed air
Soft thermoplastics	W/GT 50	HSS	130	16 – 40	3	Water: Compressed air
Hardboard and the like	W/H*	HSS	130°	16 – 25	3	Dry: Compressed air
Eternit, slate, marble	H	HSS (HM)	80	3 – 5	from hand	Dry: Compressed air
Graphite	N	HSS (HM)	80	3 – 5	from hand	Dry: Compressed air
Ebonite, Vulcanite	H	HSS	80	16 – 30	6	Dry: Compressed air
Perspex	H	HSS	130	16 – 25	3	Water

*) W = Drilling in the direction of the layers
H = Drilling at right angles to the layers

N.B. The foregoing recommendations hold good only if the following conditions are met with:

- uniform consistency of the material to be drilled;
- drills are to B S 328 and DIN 338;
- Gühring drills of HSS and HSCQ quality are used;
- maximum depth does not exceed 3 times the drill diameter;
- good machine condition and rigid mounting of the workpiece;
- no drilling bushes are used;
- correct quality of coolant and sufficient flow
- no excessive run-out of the machine spindle or drill

Bearing these points in mind the figures in the tables may be increased or reduced accordingly.

Feed rate scales in mm. per spindle revolution					
Scale No.:					
1	2	3	4	5	6
mm. per rev. ↓ s	mm. per rev. ↓ s	mm. per rev. ↓ s	mm. per rev. ↓ s	mm. per rev. ↓ s	mm. per rev. ↓ s
80 → 0.4	80 → 0.5	80 → 0.63	80 → 0.8	80 → 1.0	80 → 1.25
63 → 0.315	63 → 0.4	63 → 0.5	63 → 0.63	63 → 0.8	63 → 1.0
50 → 0.25	50 → 0.315	50 → 0.4	50 → 0.5	50 → 0.63	50 → 0.8
40 → 0.2	40 → 0.25	40 → 0.315	40 → 0.4	40 → 0.5	40 → 0.63
31.5 → 0.16	31.5 → 0.2	31.5 → 0.25	31.5 → 0.315	31.5 → 0.4	31.5 → 0.5
25 → 0.16	25 → 0.2	25 → 0.25	25 → 0.315	25 → 0.4	25 → 0.5
20 → 0.125	20 → 0.16	20 → 0.2	20 → 0.25	20 → 0.315	20 → 0.4
16 → 0.1	16 → 0.125	16 → 0.16	16 → 0.2	16 → 0.25	16 → 0.315
12.5 → 0.08	12.5 → 0.1	12.5 → 0.125	12.5 → 0.16	12.5 → 0.2	12.5 → 0.25
10 → 0.08	10 → 0.1	10 → 0.125	10 → 0.16	10 → 0.2	10 → 0.25
8 → 0.063	8 → 0.08	8 → 0.1	8 → 0.125	8 → 0.16	8 → 0.2
6.3 → 0.05	6.3 → 0.063	6.3 → 0.08	6.3 → 0.1	6.3 → 0.125	6.3 → 0.16
5 → 0.04	5 → 0.05	5 → 0.063	5 → 0.08	5 → 0.1	5 → 0.125
4 → 0.04	4 → 0.05	4 → 0.063	4 → 0.08	4 → 0.1	4 → 0.125
3.15 → 0.032	3.15 → 0.04	3.15 → 0.05	3.15 → 0.063	3.15 → 0.08	3.15 → 0.1
2.5 → 0.025	2.5 → 0.032	2.5 → 0.04	2.5 → 0.05	2.5 → 0.063	2.5 → 0.08
2 → 0.02	2 → 0.025	2 → 0.032	2 → 0.04	2 → 0.05	2 → 0.063
Drill dia. mm.	Drill dia. mm.	Drill dia. mm.	Drill dia. mm.	Drill dia. mm.	Drill dia. mm.

Cutting data for turning (meters per minute)
 (Reproduced by kind permission of Anderson Strathclyde PLC)

MATERIAL	SPEED FT/MIN M/MIN			FEED INS/REV MM/REV			DEPTH INS MM			GRADE	
	ROUGH	FINISH		ROUGH	FINISH		ROUGH	FINISH		ROUGH	FINISH
ALUMINUM ALLOYS	800	1600	2500	.04	.02	.008	.25	.18	.01	CG	CF
	250	500	750	1.	.5	.2	6.5	4.5	.25		
ALUMINUM CASTINGS	800	1600	2500	.04	.02	.008	.25	.18	.01	CG	CF
	250	500	750	1.	.5	.2	6.5	4.5	.25		
ALUMINUM CASTINGS, HT. TREATED	300	600	1600	.04	.02	.008	.25	.18	.01	CG	CF
	90	180	500	1.	.5	.2	6.5	4.5	.25		
BRASS	600	750	1000	.04	.02	.008	.25	.18	.01	CG	CF
	180	230	300	1.	.5	.2	6.5	4.5	.25		
BRONZE, PHOSPHOR	300	600	800	.04	.02	.008	.25	.18	.01	CG	CF
	90	180	250	1.	.5	.2	6.5	4.5	.25		
CAST IRON, ALLOY	150	350	500	.04	.02	.008	.25	.18	.01	CR	CG
	45	105	150	1.	.5	.2	6.5	4.5	.25		
CAST IRON, CHILLED 400B	30	60	100	.04	.02	.008	.25	.18	.01	CG	CG
	9	18	30	1.	.5	.2	6.5	4.5	.25		
CAST IRON, CHILLED 600B	25	50	60	.04	.02	.008	.25	.18	.01	CG	CG
	8	15	18	1.	.5	.2	6.5	4.5	.25		
CAST IRON, GREY	250	550	650	.04	.02	.008	.3	.2	.01	CR	CG
	75	165	190	1.	.5	.2	7.5	5.	.25		
CAST IRON, NODULAR, FERRITIC	150	300	500	.04	.02	.008	.2	.1	.01	CR	CG
	45	90	150	1.	.5	.2	5.	2.5	.25		
CAST IRON, NODULAR, PEARLITIC	150	300	450	.04	.02	.008	.2	.1	.01	CR	CG
	45	90	135	1.	.5	.2	5.	2.5	.25		
COPPER	600	1100	2000	.04	.02	.008	.25	.18	.01	CR	CG
	180	330	600	1.	.5	.2	6.5	4.5	.25		
FIBRE	300	500	700	.04	.02	.008	.25	.18	.01	CG	CF
	90	150	210	1.	.5	.2	6.5	4.5	.25		
HARD RUBBER, ASBESTOS	600	800	1000	.04	.02	.008	.3	.2	.01	CF	CF
	180	250	300	1.	.5	.2	7.5	5.	.25		
LEAD BRONZE, ALLOY	750	1000	1500	.04	.02	.008	.25	.18	.01	CW	CG
	230	300	450	1.	.5	.2	6.5	4.5	.25		

MATERIAL	SPEED FT/MIN M/MIN			FEED INS/REV MM/REV			DEPTH INS MM			GRADE	
	ROUGH	FINISH		ROUGH	FINISH		ROUGH	FINISH		ROUGH	FINISH
MALLEABLE IRON, LONG CHIP	150	500	650	.04	.02	.008	.3	.2	.01	M1	M1
	45	150	190	1	5	2	7.5	5	25		
MALLEABLE IRON, SHORT CHIP	250	400	600	.04	.02	.008	.3	.2	.01	M1/1	M1/1
	75	120	180	1	5	2	7.5	5	25		
PORCELAIN	50	60	80	.04	.02	.008	.1	.01	.003	CF	CF
	15	18	25	1	5	2	2.5	25	08		
RIGID PLASTICS, WOOD	600	800	1300	.04	.02	.008	.3	.2	.01	CW	CF
	180	250	390	1	5	2	7.5	5	25		
STEEL, ALLOY, ANNEALED	300	500	650	.03	.01	.004	.5	.25	.015	SG	SF
	90	150	190	.75	25	1	13	6.5	4		
STEEL, ALLOY, HARDENED 250B	250	400	600	.03	.01	.004	.3	.2	.01	SG	SF
	75	120	180	.75	25	.1	7.5	5	25		
STEEL, ALLOY, HARDENED 300B	200	300	500	.03	.01	.004	.25	.18	.01	SG	SF
	60	90	150	.75	25	.1	6.5	4.5	25		
STEEL, ALLOY, HARDENED 400B	150	250	350	.03	.01	.004	.25	.18	.01	SG	SF
	45	75	100	.75	25	.1	6.5	4.5	25		
STEEL, CARBON NORMALISED 125B	600	800	1100	.03	.01	.004	.5	.25	.015	SG	SF
	180	250	330	.75	25	1	13	6.5	4		
STEEL, CARBON NORMALISED 150B	400	650	1000	.03	.01	.004	.5	.25	.015	SG	SF
	120	190	300	.75	25	1	13	6.5	4		
STEEL, CARBON NORMALISED 250B	300	500	650	.03	.01	.004	.3	.2	.01	SG	SF
	90	150	190	.75	25	1	7.5	5	25		
STEEL, CAST 150B	200	300	500	.05	.01	.006	.5	.25	.015	SG	SF
	60	90	150	1.25	25	15	13	6.5	4		
STEEL, CAST 250B	150	250	350	.05	.01	.006	.3	.2	.01	SG	SF
	45	75	100	1.25	25	15	7.5	5	25		
STEEL, MANGANESE	60	100	200	.04	.02	.008	.3	.2	.01	CW	CG
	18	30	60	1	5	2	7.5	5	25		
STEEL, STAINLESS, AUSTENITIC	300	400	500	.08	.015	.008	.25	.18	.01	CW	CW
	90	120	150	2	4	2	6.5	4.5	25		
STEEL, STAINLESS, MARTENSITIC	300	400	600	.08	.015	.008	.25	.18	.01	SG	SF
	90	120	180	2	4	2	6.5	4.5	25		
STEEL, TOOL, HARDENED	30	60	100	.04	.02	.008	.2	.1	.01	SG	SF
	9	18	30	1	5	2	5	2.5	25		
STONE, HARD GRANITE	25	35	50	.04	.02	.008	.2	.1	.01	CR	CR
	8	10	15	1	5	2	5	2.5	25		
STONE, SOFT MARBLE	150	200	250	.04	.02	.008	.2	.1	.01	CF	CF
	45	60	75	1	5	2	5	2.5	25		

Kennametal Grade System

Three general purpose grades—KC850, K420, and K68—can machine all the jobs in most shops. Characteristics of these grades are:

KC850 is a rugged, multi-coated general purpose grade that delivers consistent results at accelerated speeds and feeds when machining steels. It is ideal for a wide range of jobs from roughing to semi-finishing.

K420 combines an optimum balance of toughness, with an ability to resist edge wear and cratering. It can handle heavy roughing of steel at low speeds, to light roughing at moderate speeds.

K68 was created specifically for machining aerospace metals such as superalloys and refractory metals, plus cast irons, non-ferrous metals and their alloys, and non-metals.

KC850, K420 and K68 are recommended as a means to minimize tool inventory and simplify grade selection. However, the nineteen grades listed in the table below will more precisely cover any metalcutting condition likely to be encountered. Properly applied, these tungsten carbide, titanium carbide, coated carbide, ceramic and cermet grades will provide optimum results.

Grade	Hardness RA	Typical Machining Applications
K090	94.5	Hot pressed cermet for medium roughing and finishing cast irons and heat-treated steels at high to moderate speeds and moderate to high chip loads.
K090	93-94	Cold pressed ceramic for roughing and finishing cast iron below 300 BHN, and steels up to 23-24 Rc at speeds up to 2000 sfm.
K185	93.5	Titanium carbide for finishing steels and cast irons at high to moderate speeds and light chip loads.
K7H	93.5	For finishing steels at higher speeds and moderate chip loads.
K5H	93.0	For finishing and light roughing of steels at moderate speeds and chip loads through light interruptions.
KC850	—	A rugged coated grade that produces consistent results at accelerated feeds and speeds when machining steels.
KC810	—	Coated for general machining of steels over a wide range of speeds in moderate roughing to semifinishing applications.
K45	92.5	The hardest of this group. General purpose grade for light roughing to semifinishing of steels at moderate speeds and chip loads, and for many low speed, light chip load applications.
K4H	92.0	For light roughing to semifinishing of steels at moderate speeds and chip loads, and for form tools and tools that must dwell.
K2S	91.5	For light to moderate roughing of steels at moderate speeds and feeds through medium interruptions.
K2884	92.0	A general purpose grade for milling steel over a wide range of conditions from moderate to heavy chip loads.
K21	91.0	For moderate to heavy roughing of steels at moderate speeds and heavy chip loads through medium interruptions where mechanical and thermal shock are encountered.
K420	91.3	For heavy roughing of steels at low to moderate speeds and heavy chip loads through interruptions where mechanical and severe thermal shocks are encountered.
KC210	—	Coated for general machining of cast iron over a wide range of speeds in moderate roughing to finishing applications.
K11	93.0	The hardest of this group. For precision finishing of cast irons, nonferrous alloys, nonmetals at high speeds and light chip loads, and for finishing many hard steels at low speeds and light chip loads.
K68	92.6	General purpose grade for light roughing to finishing of most high temperature alloys, refractory metals, cast irons, nonferrous alloys, and nonmetals at moderate speeds and chip loads through light interruptions.
K6	92.0	For moderate roughing of most high temperature alloys, cast irons, nonferrous alloys and nonmetals at moderate to low speeds and moderate to heavy chip loads through light interruptions.
K8735	92.0	For broaching and milling gray, malleable and nodular cast irons at moderate speeds and chip loads.
K1	90.0	The most shock resistant of this group. For heavy roughing of most high temperature alloys, cast irons, and nonferrous alloys at low speeds and heavy chip loads through heavy interruptions.

Note: Above grade designations are trademarks of Kennametal Inc.

Tooling grades—Metric ISO

HOYBIDE GRADE	WORKPIECE MATERIAL AND OPERATION	ISO CODE	COLOUR CODE
SG	For all general purpose steel cutting operations, with the exceptions of stainless steels, high-nickel alloys and heat and creep-resistant alloys.	P20	Blue
CG	For all machining operations on cast irons, meehanite, non-ferrous metals and short-chip alloy irons.	K20	Red
CW	For machining stainless steels, high-nickel alloys and heat and creep-resistant alloys. It can also be used as a universal grade for steels, cast irons, non-ferrous metals, plastics, wood, etc.	P30 M30 K20 K30	Yellow
GT	TiN coated, for machining steels, stainless steel, and general purpose operations on other metals at higher speeds with longer tool life.	P20 P30 P35 K20	Gold

The grades shown above are normally stocked in a wide range of sizes of square, triangular and rhombic, clamp and pin type, negative and positive rake conventional inserts.

Technical Data

Kennametal Grade Selection

Suggested grades and machining conditions for various work materials and types of cut.



Work Material	Hardness		FINISHING						LIGHT ROUGHING						HEAVY ROUGHING					
	BHN	RC	Up to 1/16" Depth Under .005 Feed			1/16" - 1/8" Depth 0.005 - .015 Feed			1/8" - 1/4" Depth 0.010 - .030 Feed			1/4" - 1/2" Depth 0.020 - .050 Feed			1/2" - 1" Depth 0.030 - .070 Feed			1" & Up Depth 0.040 - .100 Feed		
			Speed	Grade		Speed	Grade		Speed	Grade		Speed	Grade		Speed	Grade		Speed	Grade	
Free Machining Steels Plain Carbon Steels Alloy Steels and 400 & 500 Series Stainless Steels*	150		1500-2000			1000-1600			500-1200			375-500			300-450			275-400		
	200		1200-1800			900-1700		K45	450-1000			375-450			275-425			250-315		
	250	24	800-1500			700-1000			400-900			300-400			250-400			225-350		
	300	12	650-850		K7H	500-750			350-600			275-350			225-350			200-375		
	350	37	550-700			400-600			300-450			225-300			200-325			150-275		
	400	43	450-600			350-550			250-400			200-275			175-300			125-200		
	425	45	400-550			300-500			225-350			175-250								
	450	47	375-500			250-450			200-400			175-200								
	475	49	300-475			200-300			175-200											
	500	51	250-375			175-200														
	525	53	200-275																	

Coated Grades KC850 and KC810, permit increased metal removal rates by operating at the higher cutting speeds suggested for the steels in the above categories. In addition KC850 and KC810 are applied to other metals which are normally machined with the combined crater and edge-wear resistant grades.

Work Material	BHN	FINISHING						LIGHT ROUGHING						HEAVY ROUGHING					
		Up to 1/16" Depth			1/16" to 1/8" Depth			1/8" to 1/4" Depth			1/4" to 1/2" Depth			1/2" to 1" Depth			1" & Up Depth		
		Speed	Feed	Grade	Speed	Feed	Grade	Speed	Feed	Grade	Speed	Feed	Grade	Speed	Feed	Grade	Speed	Feed	Grade
Gray Cast Iron	110-220	400-550	0.010 - .020	K11 K090 K060	300-400	0.020 - .030	K68-K11	200-300	0.020 - .030	K68-K11	100-150	0.020 - .030	K68-K11	50-100	0.015 - .025	K68-K11	25-50	0.010 - .020	K68-K11
Nodular Iron	140-250	450-600	0.010 - .020	K11 K090 K060	375-425	0.020 - .030	K45-K68	200-325	0.020 - .030	K45-K68	150-250	0.020 - .030	K45-K68	100-200	0.020 - .030	K45-K68	50-100	0.015 - .025	K45-K68
Malleable Iron	110-220	350-500	0.010 - .020	K11 K090 K060	250-325	0.020 - .030	K45-K68	150-250	0.020 - .030	K45-K68	100-200	0.020 - .030	K45-K68	50-100	0.015 - .025	K45-K68	25-50	0.010 - .020	K45-K68
Chilled Cast Iron	470-650	60-80	0.008 - .012	K11 K090 K060	40-60	0.012 - .015	K68-K11	30-40	0.015 - .020	K68-K11	20-30	0.010 - .015	K68-K11	10-20	0.005 - .010	K68-K11	5-10	0.002 - .005	K68-K11

Coated Grade KC210 offers increased metal removal capabilities in cast iron machining by operating at the higher cutting speeds suggested.

Work Material	BHN	FINISHING						LIGHT ROUGHING						HEAVY ROUGHING					
		Up to 1/16" Depth			1/16" to 1/8" Depth			1/8" to 1/4" Depth			1/4" to 1/2" Depth			1/2" to 1" Depth			1" & Up Depth		
		Speed	Feed	Grade	Speed	Feed	Grade	Speed	Feed	Grade	Speed	Feed	Grade	Speed	Feed	Grade	Speed	Feed	Grade
200 & 300 Series Stainless Steels*	150-250	300-600	0.005 - .015	K68-K11	200-300	0.010 - .020	K6-K68	125-200	0.015 - .025	K6-K68	75-100	0.010 - .015	K6-K68	50-75	0.005 - .010	K6-K68	25-50	0.002 - .005	K6-K68
Ultra-High Strength Steels	200-350	400-600	0.010 - .020	K7H K090 K060	300-400	0.020 - .030	K21-K45	200-300	0.020 - .030	K21-K45	100-200	0.020 - .030	K21-K45	50-100	0.015 - .025	K21-K45	25-50	0.010 - .020	K21-K45
Maraging Steel 18% Ni	275-325	400-600	0.010 - .020	K7H K090 K060	300-400	0.015 - .020	K21-K45	200-300	0.015 - .020	K21-K45	100-200	0.015 - .020	K21-K45	50-100	0.010 - .015	K21-K45	25-50	0.005 - .010	K21-K45
Maraging Steel 25% Ni	175-225	250-500	0.010 - .020	K7H K090 K060	175-250	0.015 - .020	K21-K45	100-175	0.015 - .020	K21-K45	50-100	0.010 - .015	K21-K45	25-50	0.005 - .010	K21-K45	10-20	0.002 - .005	K21-K45
High Speed Steels	Over 250	300-400	0.005 - .010	K7H K090 K060	200-300	0.010 - .015	K45-K7H	100-200	0.010 - .015	K45-K7H	50-100	0.005 - .010	K45-K7H	25-50	0.002 - .005	K45-K7H	10-20	0.001 - .002	K45-K7H
Hot Work Tool Steels	Over 250	200-300	0.005 - .010	K7H K090 K060	100-200	0.008 - .012	K45-K7H	50-100	0.005 - .010	K45-K7H	25-50	0.002 - .005	K45-K7H	10-20	0.001 - .002	K45-K7H	5-10	0.0005 - .001	K45-K7H
Cold Work Tool Steels	Over 250	200-300	0.005 - .010	K7H K090 K060	100-200	0.008 - .012	K45-K7H	50-100	0.005 - .010	K45-K7H	25-50	0.002 - .005	K45-K7H	10-20	0.001 - .002	K45-K7H	5-10	0.0005 - .001	K45-K7H
High Manganese Steel	170-210	200-400	0.010 - .020	K7H K090 K060	100-200	0.010 - .015	K45-K7H	50-100	0.005 - .010	K45-K7H	25-50	0.002 - .005	K45-K7H	10-20	0.001 - .002	K45-K7H	5-10	0.0005 - .001	K45-K7H
Magnesium Alloys	Lo St	1200-1800	0.005 - .010	K68-K11	900-1200	0.010 - .018	K68-K11	600-900	0.012 - .020	K68-K11	300-600	0.010 - .018	K68-K11	150-300	0.008 - .012	K68-K11	75-150	0.006 - .010	K68-K11
Aluminum Alloys	Hi St	1200-1800	0.005 - .010	K68-K11	900-1200	0.010 - .018	K68-K11	600-900	0.012 - .020	K68-K11	300-600	0.010 - .018	K68-K11	150-300	0.008 - .012	K68-K11	75-150	0.006 - .010	K68-K11
Titanium, Pure	110-275	300-450	0.005 - .012	K68-K11	175-300	0.010 - .020	K68-K11	100-200	0.008 - .012	K68-K11	50-100	0.005 - .010	K68-K11	25-50	0.002 - .005	K68-K11	10-20	0.001 - .002	K68-K11
Titanium, Alloyed	300-380	200-300	0.005 - .010	K68-K11	150-200	0.008 - .012	K68-K11	75-150	0.005 - .010	K68-K11	30-60	0.002 - .005	K68-K11	10-20	0.001 - .002	K68-K11	5-10	0.0005 - .001	K68-K11
Copper Alloys	20-60 RB	900-1200	0.005 - .010	K68-K11	800-1000	0.008 - .012	K68-K11	600-800	0.010 - .018	K68-K11	300-600	0.008 - .012	K68-K11	150-300	0.006 - .010	K68-K11	75-150	0.005 - .008	K68-K11
Brass & Bronze	60-100 RB	800-1000	0.004 - .008	K68-K11	600-800	0.007 - .010	K68-K11	400-600	0.006 - .010	K68-K11	200-400	0.005 - .008	K68-K11	100-200	0.004 - .006	K68-K11	50-100	0.003 - .005	K68-K11
S Monel	to 200	500-1000	0.002 - .010	K68-K11	300-600	0.008 - .012	K68-K11	150-300	0.006 - .010	K68-K11	75-150	0.004 - .006	K68-K11	30-60	0.002 - .005	K68-K11	10-20	0.001 - .002	K68-K11
Inconel	to 320	250-300	0.003 - .008	K68-K11	175-275	0.008 - .012	K68-K11	100-200	0.006 - .010	K68-K11	50-100	0.004 - .006	K68-K11	25-50	0.002 - .005	K68-K11	10-20	0.001 - .002	K68-K11
W Monel	to 250	250-300	0.005 - .010	K45-K7H	175-275	0.008 - .012	K45-K7H	100-200	0.006 - .010	K45-K7H	50-100	0.004 - .006	K45-K7H	25-50	0.002 - .005	K45-K7H	10-20	0.001 - .002	K45-K7H
Zinc Alloys	Cast	500-1000	0.002 - .008	K68-K11	250-600	0.010 - .015	K68-K11	100-200	0.008 - .012	K68-K11	50-100	0.006 - .010	K68-K11	25-50	0.004 - .006	K68-K11	10-20	0.002 - .005	K68-K11
Zirconium	140-280	300-400	0.005 - .008	K68-K11	200-300	0.008 - .012	K68-K11	100-200	0.006 - .010	K68-K11	50-100	0.004 - .006	K68-K11	25-50	0.002 - .005	K68-K11	10-20	0.001 - .002	K68-K11
Manganese	140-220	275-325	0.005 - .008	K68-K11	200-275	0.008 - .012	K68-K11	100-200	0.006 - .010	K68-K11	50-100	0.004 - .006	K68-K11	25-50	0.002 - .005	K68-K11	10-20	0.001 - .002	K68-K11
Thermoplastics	600-1100	0.002 - .005		K11 K090 K060	200-500	0.006 - .012	K6-K68			K6-K68			K6-K68			K6-K68			K6-K68
Thermosetting Plastics	900-1800	0.002 - .009		K11 K090 K060	500-1000	0.008 - .015	K6-K68			K6-K68			K6-K68			K6-K68			K6-K68

*Detailed machining information on specific stainless steels is contained in Kennametal publication A78-275.

Note: Above grade designations are trademarks of Kennametal Inc.

Ceramic technical data

application considerations for successful machining of high temperature alloys

Productivity 10:1 vs carbide

When machining high temperature alloys with Kennametal ceramic cutting tools, significant productivity and machining cost advantages can be realized. There are several important factors which tooling engineers must consider to optimize their particular applications.

1. depth of cut notching (d.o.c. notching)

Notching (exaggerated abrasion wear concentrated at the depth of cut intersection of the insert) is the most common mode of failure for ceramic tools when machining high temperature alloys. Excessive d.o.c. notching can lead to premature tool failure and insert breakage.

to minimize d.o.c. notching:

- + use round inserts whenever possible
- + use the longest lead angle possible
 - when using round inserts, lead angle is optimized when d.o.c. = 5% - 15% of the diameter of the insert.
 - maximum d.o.c. = 25% of the diameter of the insert.
- + use flood coolant (no interference from clamps etc)
- + program the optimum tool path for ceramic tools.
 - maximize the effective tool lead angle.
 - prechamfer the workpiece to eliminate stress points on the cutting edge of the insert. *When using a separate prechamfer operation, feed the insert at a 90° angle to the chamfer to minimize notching of the insert during the prechamfer.*
 - avoid double notching [ex: using an 80° or 55° insert to profile in two directions with the same nose radius can lead to a double notch situation and premature tool failure.]

2. edge chipping

possible cause	solution
- wrong grade for application	- see recommendations
- speed too low	- see recommendations
- feed too high	- see recommendations
- incorrect edge prep.	- see recommendations
- vibration	- verify workpiece and tool rigidity; reduce spindle and/or tool overhang.
- interruptions in cut	- increase edge prep.
	- increase speed
	- decrease feed
	- switch to thicker insert
	- verify rigidity as above

3. excessive wear

possible cause	solution
- wrong grade for application	- see recommendations
- wrong speed (to high or low)	- see recommendations
- wrong edge prep.	- see recommendations

4. insert breakage

possible cause	solution
- improper clamping	- use carbide chip-breaker to distribute the clamping forces
	- do not over torque the lock pin when using Kenloc style inserts (12 inch/lbs. max)
- scale and/or heavy interruptions on workpiece	- increase edge prep.
- thermal shock	- and/or insert thickness
	- remove any coolant flow restrictions, must have flood coolant at insert / workpiece interface

5. coolant

The use of unrestricted flood coolant is recommended when machining high temperature alloy materials with Kennametal ceramics.

6. edge preparation

- standard edge preparation:

$$\begin{aligned} \text{Kyon 2500} &= .002 - .004 \times 20^\circ \text{ T-land (T)} \\ \text{Kyon 2000} &= .002 - .004 \text{ hone only} \\ &\text{or } .007 - .009 \times 20^\circ \text{ T-land (T)} \end{aligned}$$

These standard edge preparations will be suitable for the majority of applications machining nickel base alloys. Kennametal realizes however, that the need may arise for special or non-stocked edge preparations. Please refer to page 101 for edge preparation specifications and recommendations.



Ceramic technical data

Ceramic application recommendations for nickel and iron base high temperature alloys

work piece material	machining conditions	finishing - light rough, turn, bore	roughing - turning, boring	milling
nickel and iron base high temperature alloys machinability index 15 - 30 hardness to 38 Rc (popular examples)	depth of cut (in.)	up to .150	up to .250	up to .250
	feed rate (ipr)	.003 - .010	.005 - .012	(ipt) .004 - .012
	insert geometry	positive and negative Kendex Kenloc (Kyon2000 only)		SNC shear angle Kendex
	grade	surface speeds (sfm)		
Inco - 625	Kyon 2500 Kyon 2000	700 - 1400	400 - 800	1500 - 2000
Inco - 718	Kyon 2500 Kyon 2000	600 - 1250	400 - 800	1200 - 1600
Inco X-750	Kyon 2500 Kyon 2000	700-1250	400 - 1000	1200 - 1600
Inco - 901	Kyon 2500 Kyon 2000	400 - 1000	300 - 800	1200 - 1500
IN - 100	Kyon 2500 Kyon 2000	250 - 600	200 - 400	---
Rene 95	Kyon 2500 Kyon 2000	200 - 600	200 - 400	---
Hastelloy X	Kyon 2500 Kyon 2000	600 - 1000	400 - 800	1000 - 1400
Waspalloy	Kyon 2500 Kyon 2000	500 - 1200	400 - 1000	1200 - 1600
A - 286	Kyon 2500 Kyon 2000	500 - 800	300 - 600	---

work piece material	machining conditions	finishing - light rough, turn, bore	roughing - turning, boring	milling
cobalt based high temperature alloys (popular examples)	depth of cut (in.)	up to .060	---	up to .060
	feed rate (ipr)	.004 - .012	---	(ipt) .003 - .008
	insert geometry	positive and negative Kendex	---	negative Kendex
	grade	surface speeds (sfm)		
Haynes Alloy 25 Stellite	K090	600 - 1200	---	800 - 1500

Ceramic technical data
Ceramic application considerations for successful
machining of carbon, alloy, tool and stainless steels

1. edge preparation

A "T- land" is recommended for normal applications of Kennametal ceramic cutting tools in steel work-piece materials.

When turning and boring:
feed rate (ipr) = 1 - 1.5 x leg length (L)
of T-land.

example: feed rate = 1 - 1.5 x (.008)
[std. K090 leg length]
feed rate = .008 - .012 ipr

When milling:
feed rate (ipt) = .75 - 1 x leg length (L)
of T-land
example: feed rate = .75 - 1 x (.008)
[std. K090 leg length]
feed rate = .006 - .008 ipt



NOTE: standard edge preparation for K060 / K090 is a .008 x 20° T- land. The need may arise for special or non-stocked edge preparations. Please see page 486 for edge preparation specifications and recommendations.

2. coolant

The use of coolant is not recommended for turning, boring and milling of steels with Kennametal ceramic cutting tools.

3. edge chipping

possible cause	solution
- wrong grade for application - speed to low - feed to high - incorrect edge prep - vibrations	- see recommendations - see recommendations - see recommendations - see recommendations - check rigidity of tool & workpiece; reduce tool and/or spindle overhang.
- interruptions in cut	- increase edge prep. - increase speed - decrease feed - use thicker insert - verify rigidity as above

4. excessive wear

possible cause	solution
- wrong grade for application - speed to high or low - wrong edge preparation	- see recommendations - see recommendations - see recommendations

5. insert breakage

possible cause	solution
- improper clamping	- use carbide chip-breaker to distribute clamping forces. - do not over torque the lock pin when using Kenloc inserts (12 in/lbs maximum)
- scale and/or heavy interruptions on workpiece	- increase insert thickness and/or edge preparation
- thermal shock	- turn off coolant

Ceramic application recommendations for carbon, alloy, tool and stainless steels

work piece material	machining conditions	finishing & semi-rough, turn & bore	roughing - turning, boring	milling
free machining carbon steels AISI 1100 and 1200 series machinability index 80 - 100 hardness 140 - 190 BHN	depth of cut (in.)	up to .060	up to .150	up to .300
	feed rate (ipr)	.005 - .010	.007 - .015	.010 - .030
	insert geometry	positive & negative Kendex negative Kenloc	---	negative Kendex
	grade	surface speeds (sfm)		
	K060 K090	1000 - 3000 1000 - 3000	---	1200 - 3000

*See footnote on p. 477



Ceramic technical data

Ceramic application recommendations for carbon, alloy, tool and stainless steels (cont.)

work piece material	machining conditions	finishing & semi-rough, turn & bore		roughing - turning, boring	milling
plain carbon steel AISI 1000 series machinability index 80 - 100 hardness 180 - 240 BHN	depth of cut (in.)	up to .060	up to .150	up to .300	up to .150
	feed rate (ipr)	.005 - .010	.007 - .015	.010 - .030	(ipt) .003 - .006
	insert geometry	positive & negative Kindex negative Kenloc		---	negative Kindex
	grade	surface speeds (sfm)			
	K060 K090	1000 - 1500 1000 - 1500		---	1200 - 2000
alloy steels AISI 1300, 4000, 5000 6000, 8000 & 9000 series machinability index 80 - 100 hardness 190 - 330 BHN 330 - 450 BHN 450 - 700 BHN	K090 K090 K090	500 - 1500 500 - 1200 300 - 700		---	800 - 2000 700 - 1800 450 - 1500
tool steels wrought high speed, shock resistant, hot & cold worked tool steels machinability index 40 - 60 hardness 200 - 330 BHN 330 - 450 BHN 450 - 700 BHN	K090 K090 K090	500 - 1500 500 - 1200 300 - 700		---	800 - 2000 700 - 1500 400 - 700
stainless steels Austenitic stainless 200 - 300 series machinability index 35 - 50 hardness 140 - 190 BHN	K090	600 - 1200		---	900 - 1800
martensitic & ferritic stainless 400 - 500 & PH series machinability index 45 - 55 hardness 175 - 210 BHN	K090	650 - 1500		---	900 - 2000

* A special consideration: K090 roll turning inserts, because of their large size, thickness, and special K-lands, can handle feed rates and depths of cut not normally associated with ceramic applications

Ceramic technical data

Ceramic application considerations for successful machining of cast irons

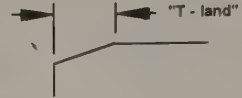
1. edge preparation

When turning and boring:
feed rate (ipr) = $1 - 3 \times \text{leg length (L) of T-land}$

example: feed rate = $1 - 3 \times (.008)$
[std. Kyon 3000 leg length]
feed rate = **.006-.024 ipr**

When milling:
feed rate (ipt) = $.75 - 1.5 \times \text{leg length (L) of T-land}$

example: feed rate = $.75 - 1.5 \times (.008)$
[std. Kyon 3000 leg length]
feed rate = **.006-.012 ipt**



NOTE: standard edge preparation for Kyon 3000, K090 and K060 is a .008 x 20° T-land. The need may arise for special or non-stock edge preparations. Please see page 486 for edge preparation specifications and recommendations.

2. coolant

Coolant is not recommended when using K060 or K090 for cast iron applications. However, coolant can safely be used with Kyon 3000 in cast iron applications and can be an effective means of dust control.

3. edge chipping

possible cause	solution
- wrong grade for application - speed to low - feed to high - incorrect edge prep - vibrations	- see recommendations - see recommendations - see recommendations - see recommendations - check rigidity of tool & workpiece; reduce tool and/or spindle overhang.
- interruptions in cut	- increase edge prep. - increase speed - decrease feed - use thicker insert - verify rigidity as above

4. excessive wear

possible cause	solution
- wrong grade for application - speed to high or low - wrong edge preparation	- see recommendations - see recommendations - see recommendations

5. insert breakage

possible cause	solution
- improper clamping	- use carbide chip-breaker to distribute clamping forces. - do not over torque the lock pin when using Kenloc inserts (12 in/lbs maximum)
- scale and/or heavy interruptions on workpiece	- increase insert thickness and/or edge preparation
- thermal shock	- turn off coolant

Ceramic application recommendations for cast iron

work piece material	machining conditions	finishing	roughing	heavy roughing	milling
cast irons grey cast iron and all other cast irons that produce a discontinuous chip machinability index 68 - 78 hardness 190 - 330 BHN 330 - 450 BHN 450 - 700 BHN	depth of cut (in.)	up to .150	up to .500	up to .500	up to .150
	feed rate (ipr)	.005 - .020	.015 - .030	up to .040	(ipt) .003 - .012
	insert geometry	positive and negative Kendex and Kenloc		neg. Kendex and Kenloc	pos. / neg. Kendex & Kenloc
	grade	surface speeds (sfm)			
	K060 K090 Kyon 3000	1000-3500 1000-3500 600-3500	1000-3500 600-3500 700-2000	600 - 3500 600 - 2000	800 - 4500 1000 - 3000
	K090 Kyon 3000	700-2000 600-2000	700-2000 600-2000	600 - 2000	1000 - 3000
	K090 Kyon 3000	500-1200 500-1500	500-2000 500-1500	500 - 1500	500 - 1500



Ceramic technical data
 Ceramic application recommendations for cast iron (cont.)

work piece material	machining conditions	finishing	roughing	heavy roughing	cutting
cast irons alloy and ductile cast irons that produce a curled chip hardness 140 - 260 BHN	depth of cut (in.)	up to .150	up to .500	up to .500	up to .150
	feed rate (ipr)	.005 - .020	.015 - .030	up to .040	(ipt) .003 - .012
	insert geometry	positive and negative Kendex and Kenloc		neg. Kendex and Kenloc	pos. / neg. Kendex & Kenloc
	grade	surface speeds (sfm)			
	K060 K090 Kyon 3000	1000 - 2000 1000 - 2000	1000 - 2000 800 - 1500	800 - 1500	1000 - 3000



Cermet technical data

considerations for successful application of Kennametal TiC /TiN Cermets

1. edge preparation

The standard edge preparations for Kennametal cermets are as follows;

insert style	grade	edge prep.
Kenloc chip control inserts	KT125 KT150 KT175	.001 - .002 hone
positive and negative Kendex turning inserts	KT125 KT150	.002 - .004 x 20° T-land
milling geometries	KT150 KT175	I.C. < 1/2" .003 - .005 x 20° I.C. ≥ 1/2" .005 - .007 x 20°

NOTE: the need may arise for special or non-stock edge preparations. Please see page 486 for edge preparation specifications and recommendations.

2. workpiece materials

Carbon, alloy, stainless steels and malleable (ductile) irons are the primary materials for TiC/TiN applications. [Free machining aluminum and non-ferrous alloys (copper, zinc, and brass) can also be machined successfully.] Note that these are all materials which machine with a ductile chip; to take full advantage of the increased chemical stability that Kennametal cermets offer.

Grey cast iron is not a primary workpiece material for cermets. The abrasion resistance required makes silicon nitride and alumina ceramics a more suitable choice as a cutting tool for grey cast iron.

3. workpiece material hardness - 40 Rc max.

TiC / TiN cermets have metallic binders, and as such do not exhibit the high hot hardness necessary for successful machining of hardened workpiece materials. >40 Rc is better suited for ceramic or CBN cutting tool materials.

4. speeds and feeds

Kennametal cermets do offer potential increased productivity vs. uncoated and coated carbides. This increased productivity will mostly be due to increased speeds.

workpiece material	speed potential (sfm) vs. uncoated carbide
carbon, stainless steels ductile irons	+30% - 50% (approx.)
alloy steels	+20% (approx.)

Cermets are more feed and fracture sensitive than uncoated or coated carbide. The current practical upper feed limit for a tough cermet (KT175) is approximately .022 - .025 ipr turning carbon steel.

The majority of successful cermet applications (turning, boring and milling) will be at conventional carbide speeds and feeds. The criteria for success will be extended tool life, superior surface finish and economics of application.

5. coolant

TiC / TiN cermets are more sensitive to thermal shock than uncoated or coated carbide. Therefore, as a general rule, coolant is not recommended when rough turning, boring, or milling.

There are cases however, as in very light finish turning and boring, when the use of coolant can be beneficial to workpiece surface finish.

6. edge chipping

possible cause	solution
- wrong grade for application	- see recommendations
- speed to low	- see recommendations
- feed to high	- see recommendations
- incorrect edge prep	- see recommendations
- vibrations	- check rigidity of tool and workpiece; reduce tool and/or spindle overhang.
- interruptions in cut	- increase edge prep - increase speed - decrease feed - use thicker insert - verify rigidity as above

7. excessive wear

possible cause	solution
- wrong grade for application	- see recommendations
- speed to high or low	- see recommendations
- incorrect edge prep	- see recommendations

8. insert breakage

possible cause	solution
- improper clamping	- do not over torque the lock pin when using Kenloc inserts. (12 in/lbs maximum)
- scale and/or heavy interruptions on workpiece	- increase insert thickness and/or edge preparation.
- thermal shock	- turn coolant off

Cermet technical data

application recommendations

		grade	surface speed (sfm)	depth of cut (in)	feed rate (in./rev.)
free machining and plain carbon steels 140 - 240 BHN	finishing	KT125	300 - 1500	.100 max.	.015 max.
	semi-finish/ light roughing	KT150 / KT175	150 - 1000	.150 max.	.020 max.
	milling	KT175	300 - 1000	as required	.002 - .010 ipt
alloy steels -190 - 330 BHN	finishing	KT125	300 - 1200	.100 max.	.015 max.
	semi-finish/ light roughing	KT150 / KT175	150 - 800	.150 max.	.020 max.
	milling	KT175	300 - 800	as required	.002 - .010 ipt
alloy steels 330 - 400 BHN	finishing	KT125	300 - 800	.100 max.	.015 max.
	semi-finish/ light roughing	KT150 / KT175	100 - 500	.150 max.	.020 max.
	milling	KT175	200 - 600	as required	.002 - .010 ipt
stainless steel austenitic 200 - 300 series 140 - 190 BHN	finishing	KT125	300 - 1200	.100 max.	.015 max.
	semi-finish/ light roughing	KT150 / KT175	300 - 1000	.150 max.	.015 max.
	milling	KT175	300 - 1000	as required	.002 - .010 ipt
stainless steel martensitic 400 - 500 PH series 175 - 210 BHN	finishing	KT125	600 - 1000	.100 max.	.015 max.
	semi-finish/ light roughing	KT150 / KT175	600 - 800	.150 max.	.015 max.
	milling	KT175	400 - 800	as required	.002 - .010 ipt
alloy cast iron (ductile, malleable)	finishing	KT125	300 - 1500	.125 max.	.015 max.
	semi-finish/ light roughing	KT150 / KT175	200 - 1200	.250 max.	.020 max.
	milling	KT175	300 - 1500	as required	.002 - .015 ipt
free machining aluminum and non ferrous alloys (copper, zinc, brass)	finishing	KT125	1000 - 4000	.125 max.	.015 max.
	semi-finish/ light roughing	KT150 / KT175	700 - 4000	.250 max.	.020 max.
	milling	KT175	500 - 3000	as required	.002 - .015 ipt



Polycrystalline technical data

Recommendations for successful application of Kennametal Polycrystalline diamond and CBN tools

1. edge preparation

The standard edge preparations for Kennametal polycrystalline inserts are as follows:

insert style	grade	edge prep.
positive Kendex	all KD100 PCD grades	sharp
negative Kendex	KD100 KD120 KD050 KD220 KD200	sharp .001 - .002 hone .004 x 20° T-land .004 x 20° T-land .008 x 20° T-land & .001 - .002 hone
negative Kenloc	KD100 KD120 KD050 KD220	sharp .001 - .002 hone .004 x 20° T-land .004 x 20° T-land
Top Notch threading and grooving	KD100	sharp
Top Notch profiling	KD120	.001 - .002 hone
positive screw-on mini screw-on brazed tools	all KD100 PCD grades	sharp

NOTE: the need may arise for special or non-stock edge preparations. See page 486 for edge preparation specifications and recommendations

2. tool geometries

When applying KD100 (PCD) in aluminum or other non-ferrous applications, it is beneficial to use a positive tool geometry to minimize cutting pressure and eliminate built-up edge.

When applying KD050, KD120, KD220 or KD200 to machine hardened ferrous alloys, it is beneficial to use a negative tool geometry to maximize insert strength, stabilize the cutting edge and improve tool life. This can be valid even in situations where the depth of cut is very low ($< .010$).

3. workpiece material hardness (CBN applications)

When machining hardened ferrous materials with Kennametal CBN's, maximum economic benefits can be realized if the workpiece hardness is 45 - 65 Rc. An exception to this is the sintered irons which are relatively soft (200 - 250 BHN). The abrasiveness of these irons however, warrant the use of CBN tools.

4. coolant

The use of flood coolant is strongly recommended when applying KD050, KD120 and KD220. Coolant is not recommended when applying KD200 (solid CBN), nor is it necessary when applying KD100 in nonferrous applications.

5. edge chipping

possible cause	solution
- wrong grade for application	- see recommendations
- speed to low	- see recommendations
- feed to high	- see recommendations
- incorrect edge prep.	- see recommendations
- vibrations	- check rigidity of tool and workpiece; reduce tool and/or spindle overhang
- interruptions in cut	- increase edge prep. - increase speed - decrease feed - use thicker insert - verify rigidity as above

6. excessive wear

possible cause	solution
- wrong grade for application	- see recommendations
- speed to high or low	- see recommendations
- incorrect edge prep.	- see recommendations

7. insert breakage

possible cause	solution
- improper clamping	- do not over torque the lock pin when using Kenloc inserts (12 in/lbs max.)
- scale and/or heavy interruptions on workpiece	- increase insert thickness and/or edge preparation

8. Polycrystalline tip dislodged

possible cause	solution
- braze wetting between carbide and polycrystalline tip.	- flood coolant to tip of insert - lower speed; see recommendations.

Polycrystalline diamond technical data

application recommendations

non-ferrous materials		grade	surface speed (sfm)	depth of cut (doc)	feed rate in / rev.
free machining aluminum alloys 80 - 120 BHN	finishing	KD100	2000-4000	to .060"	.005-.010
	roughing	KD100	1800-3500	to .125"	.008-.020
	milling	KD100	1000-10,000	to .125"	.003-.010 ipt
high silicon aluminum hypereutectic	finishing	KD100	2000-3000	to .060"	.005-.010
	roughing	KD100	1200-2500	to .125"	.008-.020
	milling	KD100	1200-3000	to .125"	.003-.010 ipt
nonferrous free machining alloys copper, zinc, brass alloys 80 - 120 BHN	finishing	KD100	1200-3500	to .060"	.005-.010
	roughing	KD100	1200-3000	to .125"	.008-.020
	milling	KD100	1800-3500	to .125"	.003-.010 ipt
non- metallics nylons, acrylics, phenolic resin materials	finishing	KD100	2000-4500	to .060"	.005-.010
	roughing	KD100	550-2500	to .125"	.008-.020
	milling	KD100	1000-4500	to .125"	.003-.010 ipt



Polycrystalline CBN technical data **application recommendations**

hardened steels		grade	surface speed (sfm)	depth of cut (doc)	feed rate in/rev.
cold work tool steel 570-750 BHN 55-65 Rc	finishing	KD050	250 - 500	.050" max.	.002 - .008
	semi-finishing / light roughing	KD120 KD220	200 - 450 200 - 350	.100" max. .125" max.	.002 - .010 .002 - .012
	rough turning	KD200	200 - 350	.150" max.	.002 - .012
	milling		500 - 700	.150" max.	.006 - .010 (ipt)
high-speed steel 570-750 BHN 55-65 Rc	finishing	KD050	250 - 500	.050" max.	.002 - .008
	semi-finishing / light roughing	KD120 KD220	200 - 450 200 - 350	.100" max. .125" max.	.002 - .010 .002 - .012
	roughing	KD200	200 - 350	.150" max.	.002 - .012
bearing steel 570-750 BHN 55-65 Rc	finishing	KD050	250 - 500	.050" max.	.002 - .008
	semi-finishing / light roughing	KD120	200 - 450	.100" max.	.002 - .010
	roughing	---	---	---	---
hot work die steel 485-650 BHN 50-60 Rc	finishing	KD050	250 - 500	.050" max.	.002 - .008
	semi-finishing / light roughing	KD120	200 - 450	.100" max.	.002 - .010
	roughing	---	---	---	---
surface hardened steel 570-750 BHN 56-65 Rc	finishing	KD050	250 - 500	.050" max.	.002 - .008

Polycrystalline CBN technical data

application recommendations

hardened cast irons		grade	surface speed (sfm)	depth of cut (doc)	feed rate in/rev.
white chilled iron 485-650 BHN 50-60 Rc	finishing	KD120	250 - 450	.060" max.	.002 - .006
	semi-finishing / light roughing	KD220	200 - 350	.100" max.	.005 - .010
	rough turning	KD200	200 - 350	.150" max.	.008 - .016"
	milling		500 - 1000	.150" max.*	.006 - .015 (ipt)*
high chromium iron 570-750 BHN 55-65 Rc	finishing	KD120	100 - 200	.050" max.	.002 - .005
	semi-finishing / light roughing	KD220	100 - 200	.080" max.	.002 - .008
	roughing	KD200	100 - 200	.125" max.	.005 - .010
Ni Hard 570-750 BHN 55-65 Rc	finishing	KD120	125 - 250	.050" max.	.002 - .006
	semi-finishing / light roughing	KD220	125 - 250	.080" max.	.002 - .008
	roughing	KD200	125 - 250	.125" "	.005 - .015"
	milling		650 - 1000		.008 - .012 (ipt)*
sintered Iron 200-250 BHN	finishing	KD120	800 - 1000	.050" max.	.002 - .006
	semi-finishing / light roughing	KD220	700 - 1000	.050" max.	.002 - .006

* Use round chamfered inserts.

high temperature alloys

high temperature nickle base alloys 325-500 BHN 35-50 Rc	finishing	KD120	400 - 600	.080" max.	.003 - .007
high temperature cobalt base hard facing alloys 570-750 BHN 55-65 Rc	finishing	KD120	600 - 800	.050" max.	.004 - .008
	semi-finishing / light roughing	KD220	600 - 800	.050" max.	.004 - .010
	roughing	KD200	600 - 800	.050" max.	.004 - .012



technical data

edge preparation

Ceramics, cermets and polycrystalline cutting tool materials have a relatively high material hardness. Because of this, the cutting edge is more brittle than the conventional carbide metal cutting tools.

In order to optimize the performance of these advanced cutting tool materials, it is critical that the edge preparation be matched to the cutting tool material, the workpiece material and the machining operation being performed.

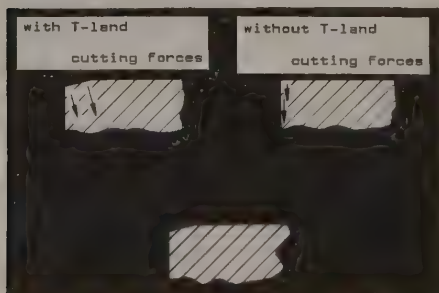
General recommendations for standard edge preparations by cutting tool material and machining conditions have been given in each of the three sections of this catalog.

There are four basic choices when considering edge preparation for advanced materials:

1. T - land
2. hone
3. T - land plus hone
4. up - sharp

1. T - land

T-lands act to protect the cutting edge by eliminating a sharp cutting edge, thereby reducing edge chipping, and by redistributing the cutting forces back through the larger body cross section of the insert which makes the insert stronger.



typical T-lands

angle 'A'	width 'W'
20° / 25°	.002 - .004
	.005 - .007
	.007 - .009
	.012 - .014
30° / 35°	.005 - .007
	.008 - .010
	.012 - .014
10° / 15°	.014 - .016
	.030 - .032
	.060 - .062

There is a trade off. Increasing either the width 'W' or the angle 'A' of the T-land to strengthen the insert will also increase the cutting forces and could have a potentially detrimental effect on both the cutting tool and/or the workpiece material.

The optimum situation is to have the minimum edge preparation (in terms of T-land size and angle) which will perform the desired operation, and protect the insert cutting edge adequately, thereby achieving satisfactory tool life.

2. hone

Hones protect the cutting edge by eliminating the sharp edge and reducing edge chipping. Hones do not provide the high level of edge protection of a T-land. However, they can be very satisfactory when using ceramics, cermets or polycrystalline inserts in finishing operations, such as when depth of cut and feed rates are kept at a minimum, and minimum cutting pressure is desired.

Kennametal recommends the following range of radius type hones:

A radius hone	
hone designation	'R' radius
A	.001 - .002
B	.003 - .004
C	.005 - .006

3. T-land plus hone

In some situations, like rough turning steel with ceramics, minute chipping can occur at the intersections of the T-land and the rake surface or flank surface of the insert. This condition can be corrected by applying a radius hone to both intersections. This can effectively improve the performance of the edge preparation while keeping T-land size and angle constant.

Kennametal recommends the following modifications:

A radius hone	
hone designation	'R' radius
A	.001 - .002
B	.003 - .004
C	.005 - .006

technical data

edge preparation

4. up - sharp

Normally, Kennametal does not recommend up-sharp edges for ceramic, cermet or polycrystalline inserts. The high material hardness results in an edge that is highly susceptible to chipping if left 'up-sharp'.

However, if it has been determined that an up-sharp edge is necessary, it can be ordered as a special.

how to order an Insert with a non-standard edge preparation:

When ordering any Kennametal advanced cutting tool material with a non-standard edge preparation, please detail the type and size of edge preparation desired.

ex: CNGA-432T KY3000 with .012 x 30° T-land

When considering special edge preparations, please contact your local Kennametal Tooling Systems Engineer. He is uniquely qualified to make recommendations to "fine tune" your advanced cutting tool applications, and to help you maximize your metalcutting productivity.

metalcutting safety

Modern metalcutting operations involve high energy, high spindle or cutter speeds, and high temperatures and cutting forces. Hot, flying chips may be projected from the workpiece during metalcutting. Although advanced cutting tool materials are designed and manufactured to withstand the high cutting forces and temperatures that normally occur in these operations, they are susceptible to fragmenting in service, particularly if they are subjected to over-stress, severe impact or otherwise abused. Therefore, precautions should be taken to adequately protect workers, observers and equipment against hot, flying chips, fragmented cutting tools, broken workpieces or other similar projectiles. Machines should be fully guarded and personal protective equipment should be used at all times.

When grinding advanced cutting tool materials, a suitable means for collection and disposal of dust, mist or sludge should be provided. Overexposure to dust or mist containing metallic particles can be hazardous to health particularly if exposure continues over an extended period of time and may cause eye, skin and mucous membrane irritation and temporary or permanent respiratory disease. Certain existing pulmonary and skin conditions may be aggravated by exposure to dust or mist. Adequate ventilation, respiratory protection and eye protection should be provided when grinding and workers should avoid breathing of and prolonged skin contact with dust or mist. General Industry Safety and Health Regulations, Part 1910, U.S. Department of Labor, published in Title 29 of the Code of Federal Regulations should be consulted. Read the applicable Material Safety Data Sheet before grinding.

Cutting tools are only one part of the worker-machine-tool system. Many variables exist in machining operations, including the metal removal rate; the workpiece size, shape, strength and rigidity; the chucking or fixturing; the load carrying capability of centers; the cutter and spindle speed and torque limitations; the holder and boring bar overhang; the available power; and the condition of the tooling and the machine. A safe metalcutting operation must take all of these variables, and others, into consideration.

Kennametal has no control over the end use of its products or the environment into which those products are placed. Kennametal urges that its customers adhere to the recommended standards of use of their metalcutting machines and tools, and that they follow procedures that ensure safe metalcutting operations.

For more information, we suggest you write for Kennametal's Metalcutting Safety booklet Number A86-69(10)E6, if you do not already have one. Quantities are available, free, for distribution to your operating personnel.

Tool Design Data Formula—thrust reaming—boring—core drilling

Reaming, Boring & Core Drilling Thrust
Thrust = 160,000 · K · FPR · (Width of cut)

$$\text{Thrust} = 160,000 \cdot K \cdot \text{FPR} \cdot \left(\frac{D-d}{2} \right) \times 1.5 \text{ (dull tool factor)}$$

Material	K	Material	K
Dow Metal	.20	1320 Steel	1.00
Bearing Bronze	.40	6140 Steel	1.00
Alum. Alloy	.48	1020 Steel	1.05
Cast Iron	.80	2320 Steel	1.05
Malleable C.I.	.90	3120 Steel	1.05
1112 Steel	.90	2345 Steel	1.10
1315 Steel	.95	1095 Steel	1.40
1120 Steel	.96	Ann Copper	1.15

Note: Use this formula when tool has lead angle.

Spotface and Counterboring Thrust
Thrust = 232,000 · K · FPR · (Width of cut)

$$\text{Thrust} = 232,000 \cdot K \cdot \text{FPR} \cdot \left(\frac{D-d}{2} \right) \times 1.5 \text{ (dull tool factor)}$$

Material	Brinell	K (Thrust)
Mag. H-Alloy	63	.27
Alum. Alloy 24 ST	154	.82
Alum. Alloy 2S	37	2.70
Leaded Brass	124	.21
Phos. Bronze	156	.82
Monel Metal	207	1.36
C.I. (40,000 PSI)	210	.71
1020 Steel C.D.	180	1.00
6150 Steel H.R.	195	1.72
6150 Steel H.T.	241	2.09
1095 Steel Ann.	156	3.04

Note: Use this formula when tool has no lead angle such as Spotface tool.

Tool Design Data—Formula-horsepower tapping

Tapping

$$\left(\frac{1/N \cdot \text{RPM}}{10} \right) \times \text{HP from chart} \times 1.3^* = \text{HP}$$

$N = \text{No. Thd's per inch}$

HP required for tapping various materials (approx.)

(to be used for estimating purposes only)

Note—tap drill sizes to give approx. 75% full thread

HP for various materials based on a feed of 10' per minute

Tap Size	rpm	Drill size		Class					
		Frac.	Dec.	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6
1/4 — 20	200	13/64	0.203	0.058	0.086	0.113	0.140	0.165	0.189
1/4 — 28	280	7/32	0.219	0.043	0.064	0.084	0.104	0.122	0.140
5/16 — 18	180	F	0.257	0.082	0.121	0.160	0.198	0.232	0.266
5/16 — 24	240	I	0.272	0.064	0.094	0.125	0.155	0.181	0.207
3/8 — 16	160	5/16	0.312	0.112	0.165	0.218	0.270	0.317	0.363
3/8 — 24	240	Q	0.332	0.078	0.114	0.151	0.188	0.219	0.251
7/16 — 14	140	U	0.368	0.150	0.221	0.292	0.362	0.426	0.486
7/16 — 20	200	W	0.386	0.109	0.161	0.212	0.263	0.308	0.353
1/2 — 13	130	27/64	0.422	0.185	0.272	0.360	0.447	0.523	0.600
1/2 — 20	200	29/64	0.453	0.125	0.184	0.243	0.302	0.353	0.405
9/16 — 12	120	31/64	0.484	0.227	0.334	0.441	0.548	0.641	0.735
9/16 — 18	180	33/64	0.516	0.157	0.231	0.305	0.379	0.444	0.508
5/8 — 11	110	17/32	0.531	0.275	0.405	0.534	0.664	0.777	0.890
5/8 — 18	180	37/64	0.578	0.176	0.259	0.342	0.425	0.497	0.570
3/4 — 10	100	21/32	0.656	0.366	0.539	0.711	0.883	1.034	1.184
3/4 — 16	160	11/16	0.687	0.237	0.349	0.460	0.572	0.669	0.767
7/8 — 9	90	49/64	0.766	0.477	0.702	0.926	1.151	1.347	1.544
7/8 — 14	140	13/16	0.812	0.317	0.467	0.616	0.765	0.895	1.026
1 — 8	80	7/8	0.875	0.614	0.903	1.192	1.481	1.734	1.987
1 — 14	140	15/16	0.937	0.327	0.481	0.635	0.789	0.924	1.058
1-1/8 — 7	70	63/64	0.984	0.780	1.146	1.513	1.879	2.20	2.521
1-1/8 — 12	120	1-3/64	1.047	0.518	0.762	1.006	1.250	1.463	1.676
1-1/4 — 7	70	1-7/64	1.109	0.874	1.286	1.697	2.108	2.468	2.828
1-1/2 — 6	60	1-11/32	1.343	1.238	1.821	2.404	2.986	3.500	4.00
1-1/2 — 12	120	1-27/64	1.422	0.700	1.050	1.359	1.689	1.977	2.265

Note: HP values for all other materials can be based on machinability value for tapping.

*Add 30.5% to horsepower for dull tools and machine friction.

Select class designation from following sheet

Example 1/2-13 tap 1040 material

$$\left(\frac{1/13 \cdot 425}{10} \right) \times 0.523 \times 1.3 = 2.225 \text{ HP}$$

Tool Design Data
Formula—Horsepower classification of material

Classifications of materials for computing horsepower required for tapping

Class 1	Class 4
Aluminum—Rod; Cast; Die-Cast	Iron—Malleable
Brass—Bar; Cast	Steel—1010; 1035; 1112
Zinc—Die-Cast	X-1340 Cast
Class 2	Class 5
Brass—Stamping; tubing	Iron, Wrought
Bronze, Phosphor; Tubing; Cast	Steel—1040; 1095
Bar; Maganese	T-1330—T-1350; Stamping
Cast Aluminum	
Copper; German Silver	
Class 3	Class 6
Brass—Forging	Monel Metal
Bronze—Naval	Steel—5120—5210
Iron—Cast; Nickel	6115—6195
Steel—Semi; Casting	4130—4820
	2015—2515
	3115—3450
	Forging; Nitralloy
	Stainless—Free

Determining Power Requirements in Machining†

TABLE 17.2-1 Shop Formulas for Turning, Milling, Drilling and Broaching—English Units

PARAMETER	TURNING	MILLING	DRILLING	BROACHING
Cutting speed, fpm	$V_c = .262 \times D_1 \times \text{rpm}$	$V_c = .262 \times D_m \times \text{rpm}$	$V_c = .262 \times D_d \times \text{rpm}$	V_c
Revolutions per minute	$\text{rpm} = 3.82 \times \frac{V_c}{D_1}$	$\text{rpm} = 3.82 \times \frac{V_c}{D_m}$	$\text{rpm} = 3.82 \times \frac{V_c}{D_d}$	—
Feed rate, in/min	$f_m = f_r \times \text{rpm}$	$f_m = f_r \times n \times \text{rpm}$	$f_m = f_r \times \text{rpm}$	—
Feed per tooth, in	—	$f_t = \frac{f_m}{n \times \text{rpm}}$	—	f_t
Cutting time, min	$t = \frac{L}{f_m}$	$t = \frac{L}{f_m}$	$t = \frac{L}{f_m}$	$t = \frac{L}{12 V_c}$
Rate of metal removal, in ³ /min	$Q = 12 \times d \times f_r \times V_c$	$Q = w \times d \times f_m$	$Q = \frac{\pi D_d^2}{4} \times f_m$	$Q = 12 \times w \times d_r \times V_c$
Horsepower required at spindle*	$hp_s = Q \times P$	$hp_s = Q \times P$	$hp_s = Q \times P$	—
Horsepower required at motor*	$hp_m = \frac{Q \times P}{E}$	$hp_m = \frac{Q \times P}{E}$	$hp_m = \frac{Q \times P}{E}$	$hp_m = \frac{Q \times P}{E}$
Torque at spindle	$T_s = \frac{63030 \text{ hp}_s}{\text{rpm}}$	$T_s = \frac{63030 \text{ hp}_s}{\text{rpm}}$	$T_s = \frac{63030 \text{ hp}_s}{\text{rpm}}$	—

SYMBOLS: D_1 = Diameter of workpiece in turning, inches
 D_m = Diameter of milling cutter, inches
 D_d = Diameter of drill, inches
 d = Depth of cut, inches
 d_1 = Total depth per stroke in broaching, inches
 E = Efficiency of spindle drive
 f_m = Feed rate, inches per minute
 f_r = Feed, inches per revolution
 f_t = Feed, inches per tooth
 hp_m = Horsepower at motor
 hp_s = Horsepower at spindle
 L = Length of cut, inches
 n = Number of teeth in cutter
 P = Unit power, horsepower per cubic inch per minute
 Q = Rate of metal removed, cubic inches per minute
 rpm = Revolutions per minute of work or cutter
 T_s = Torque at spindle, inch-pounds
 t = Cutting time, minutes
 V_c = Cutting speed, feet per minute
 w = Width of cut, inches

*Unit power data are given in table 17.2-3 for turning, milling and drilling, and in figure 17.2-1 for broaching.

†Reprinted by permission from *Machining Data Handbook*, 3rd edition, © 1980, Metant Research Associates, Inc.

Determining Power Requirements in Machining[†]

TABLE 17.2-3 Average Unit Power Requirements for Turning, Drilling and Milling—English Units

MATERIAL	HARDNESS Bhn	UNIT POWER ^o hp/in ^{1/2} /min					
		TURNING P _s HSS AND CARBIDE TOOLS (feed .005-.020 ipr)		DRILLING P _s HSS DRILLS (feed .002-.008 ipr)		MILLING P _s HSS AND CARBIDE TOOLS (feed .005-.012 ipt)	
		Sharp Tool	Dull Tool	Sharp Tool	Dull Tool	Sharp Tool	Dull Tool
STEELS, WROUGHT AND CAST Plain Carbon Alloy Steels Tool Steels	85-200	1.1	1.4	1.0	1.3	1.1	1.4
	35-40 R _c	1.4	1.7	1.4	1.7	1.5	1.9
	40-50 R _c	1.5	1.9	1.7	2.1	1.8	2.2
	50-55 R _c	2.0	2.5	2.1	2.6	2.1	2.6
	55-58 R _c	3.4	4.2	2.6	3.2 [*]	2.6	3.2
CAST IRONS Gray, Ductile and Malleable	110-190	0.7	0.9	1.0	1.2	0.6	0.8
	190-320	1.4	1.7	1.6	2.0	1.1	1.4
STAINLESS STEELS, WROUGHT AND CAST Ferritic, Austenitic and Martensitic	135-275	1.3	1.6	1.1	1.4	1.4	1.7
	30-45 R _c	1.4	1.7	1.2	1.5	1.5	1.9
PRECIPITATION HARDENING STAINLESS STEELS	150-450	1.4	1.7	1.2	1.5	1.5	1.9
TITANIUM	250-375	1.2	1.5	1.1	1.4	1.1	1.4
HIGH TEMPERATURE ALLOYS Nickel and Cobalt Base	200-360	2.5	3.1	2.0	2.5	2.0	2.5
	Iron Base	1.6	2.0	1.2	1.5	1.6	2.0
REFRACTORY ALLOYS Tungsten	321	2.8	3.5	2.6	3.3 [*]	2.9	3.6
Molybdenum	229	2.0	2.5	1.6	2.0	1.6	2.0
Columbium	217	1.7	2.1	1.4	1.7	1.5	1.9
Tantalum	210	2.8	3.5	2.1	2.6	2.0	2.5
NICKEL ALLOYS	80-360	2.0	2.5	1.8	2.2	1.9	2.4
ALUMINUM ALLOYS	30-150 500 kg	0.25	0.3	0.16	0.2	0.32	0.4
MAGNESIUM ALLOYS	40-90 500 kg	0.16	0.2	0.16	0.2	0.16	0.2
COPPER	80 R _B	1.0	1.2	0.9	1.1	1.0	1.2
COPPER ALLOYS	10-80 R _B	0.64	0.8	0.48	0.6	0.64	0.8
	80-100 R _B	1.0	1.2	0.8	1.0	1.0	1.2

^oPower requirements at spindle drive motor, corrected for 80% spindle drive efficiency.

^{*}Carbide

[†]Reprinted by permission from *Machining Data Handbook*, 3rd edition, ©1980, Metcut Research Associates, Inc.

APPENDIX C

PROGRAMMING EXERCISES

Ideally, any part programming exercise should be supported by program proving and production of the component. Most of the inch and metric exercises that follow will fall within the capacity of equipment now generally available in educational institutions, and it is likely that most benefit will be gained if the exercises are completed with that equipment in mind.

To cater to students who do not have access to suitable equipment two imaginary machine types follow together with their control systems. These alternative systems are fairly typical and are illustrative of the way manufacturers summarize the capabilities of machines and control systems in their promotional literature and instruction manuals.

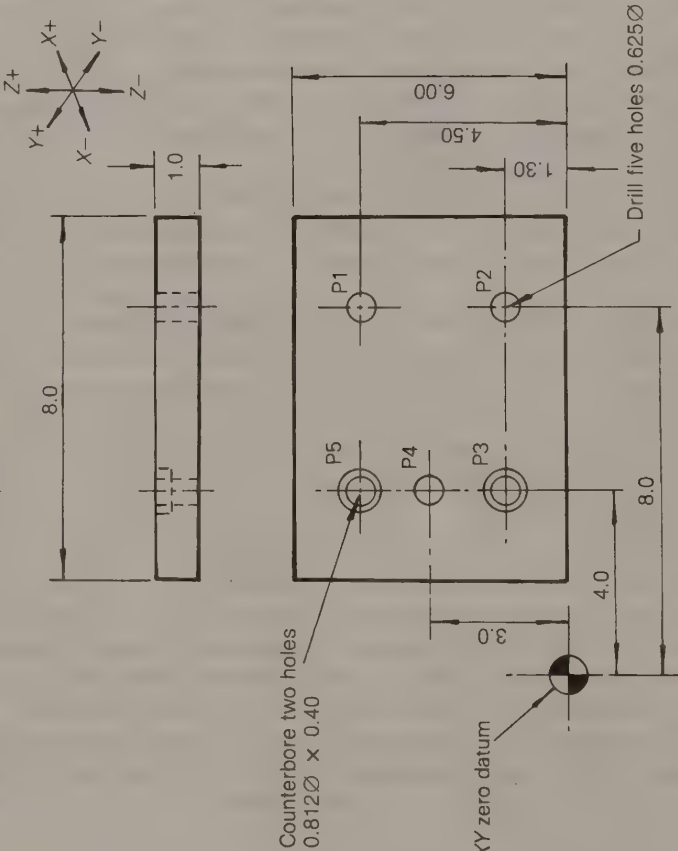
It is anticipated, if the exercises are attempted using the alternative systems, that, as when using college-based equipment, the student will receive further guidance and advice from his or her lecturer. Partially completed programs for Exercises 1, 2, 9, and 10 are included at the end of the chapter.

Details of the machining, tooling and programming requirements are included alongside each detail drawing with the exception of the last exercise of each type, where the intention is that the student should complete all stages from detail drawing to finished product, making decisions relating to work holding, tooling, feeds and speeds followed by part programming and, in cases where the programming has been related to equipment available, by program proving and final machining.

Appropriate speeds and feeds may be selected from the data given in Chapter 7, Appendix B, or from a *Machining Data Handbook*.

The components have been dimensioned according to traditional standards in some cases, and in others the method of dimensioning which is increasingly being favored for numerical control is used. Students will be able to judge for themselves which is the more appropriate conventional or coordinate.

The direction diagrams included on Exercises 1, 2, 9, and 10 indicate relative tool movement, that is, the directions shown will be those used in the part program. The diagrams relating to the turning exercise assume a rear mounted tool post.

EXERCISE NO.	1
<i>Machining</i> Center drill and drill five holes 0.625$\varnothing$ in sequence P1 to P5 Counterbore two holes 0.812$\varnothing$ x 0.40 deep in sequence P3 to P5 <i>Tooling</i> HSS No. 2 center drill HSS drill — .625$\varnothing$ HSS counterbore .812$\varnothing$ <i>Programming</i> Absolute dimensions Point-to-point positioning Linear interpolation Z datum clearance 0.100 Time dwell (to clear counterbore) Flood coolant Spindle speed, rev/min Feed inches Allocate tool offsets Do not use drill cycle	 Material: mild steel Inch Programming Exercise (see example program 1)

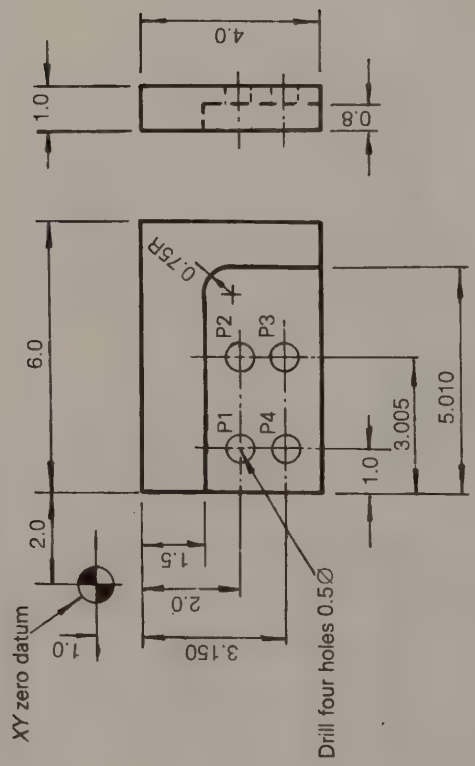
EXERCISE NO.	2
<i>Machining</i> Mill 12 slot, P1 start Center drill and drill three holes 5Ø in sequence P2 to P4	<i>Tooling</i> HSS 12Ø end mill HSS No. 2 center drill HSS drill 5Ø <i>Programming</i> Incremental dimensions Linear interpolation Point-to-point positioning Quill clamp (when milling slot) Drill cycle Spray mist coolant Spindle speed rev/min Feed mm/min Allocate tool offsets Z datum clearance 4mm

Material: mild steel

Metric Programming

Exercise

(see example program 2)

EXERCISE NO.	3
Machining Mill 0.800 step in two passes Center drill and drill four holes .5Ø sequence P1 to P4 Tooling Tungsten carbide insert end mill, 1.5Ø, four teeth HSS No. 2 center drill HSS drill 0.5Ø Programming Absolute dimensions Linear interpolation Point-to-point positioning Quill clamp (when milling steps) Flood coolant Drill cycle Spindle speed rev/min Feed inches/min Allocate tool offsets Z datum clearance 0.200	Material: brass (Inch Programming Exercise)  XY zero datum Drill four holes 0.5Ø

EXERCISE NO.	4
<i>Machining</i> Center drill, drill and ream two holes 8Ø Mill radial profile P1 to P2	<i>Tooling</i> HSS No. 2 center drill HSS drill 7.5Ø HSS reamer 8Ø Cemented carbide insert end mill 20Ø, three teeth <i>Programming</i> Absolute dimensions Point-to-point positioning Linear interpolation Circular interpolation Z datum clearance 2mm Drill cycle Ramp down inhibit Cutter compensation Flood coolant Spindle speed rev/min Feed mm/min Allocate tool offsets

Material: mild steel
(flame-cut profile)

(Metric Programming Exercise)

EXERCISE NO.	5
Machining Drill and ream 1.5Ø hole Rough mill corners and finish profile with continuous cut P1 to P6	
Tooling HSS No. 3 center drill HSS drill 0.812Ø HSS drill 1.468Ø HSS reamer 1.5Ø Cemented carbide insert end mill 1.25Ø three teeth	
Programming Absolute dimensions Z datum clearance 0.200 Linear interpolation Circular interpolation Cutter compensation Spray mist coolant Spindle speed rev/min Feed inches/rev Allocate tool offsets	

The drawing shows a mechanical part with the following dimensions and features:

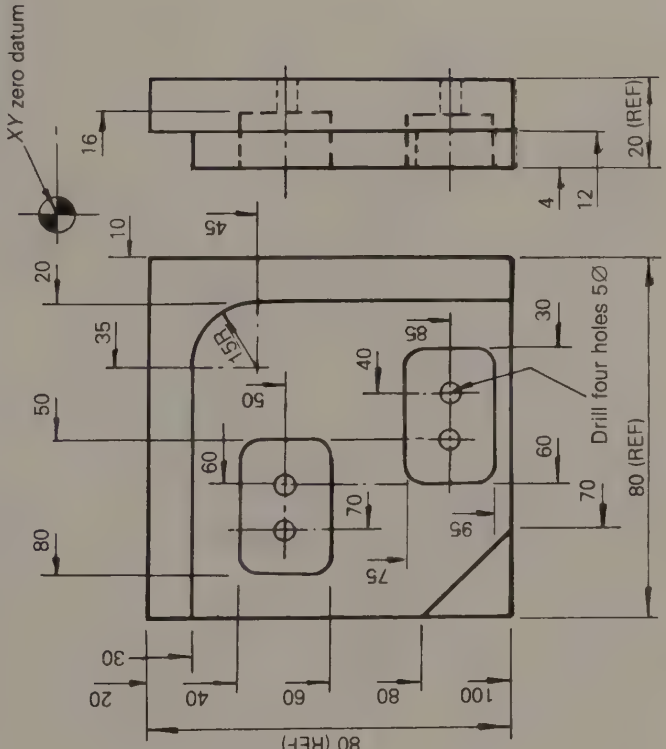
- XY zero datum:** Indicated by a circle at the top left corner.
- Overall dimensions:** 8.0 (REF) width, 8.0 (REF) depth, 1.2 height.
- Internal features:** A central hole with a diameter of 1.5Ø. A fillet with a radius of 1.5R connects the top left corner to the central hole.
- Points P1 to P6:** P1 is at the top left corner. P2 is at the top right corner. P3 is at the bottom right corner. P4 is at the bottom left corner. P5 is at the top left corner. P6 is at the bottom left corner.
- Dimensions:** 3.0, 5.0, 6.5, 8.0 (REF), 1.0 (REF), 4.0, 8.0, 2.0, 6.0, 4.0, 2.0, 6.0, 1.5, 3.0, 5.0, 7.0, 2.0R, 1.5R.
- Annotation:** "Drill and ream 1.5Ø" with an arrow pointing to the central hole.

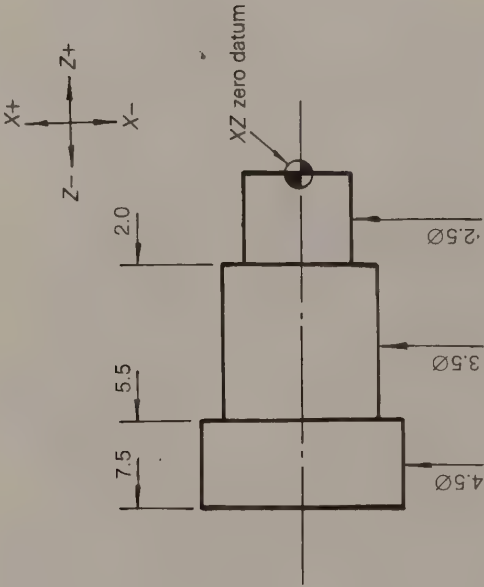
Dimensions indicated are from zero datum

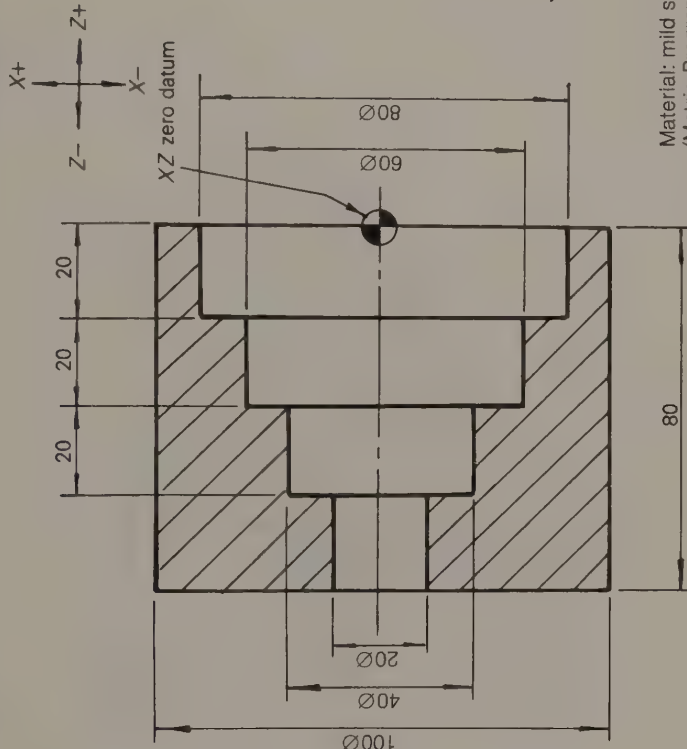
Material aluminum alloy (Inch Programming Exercise)

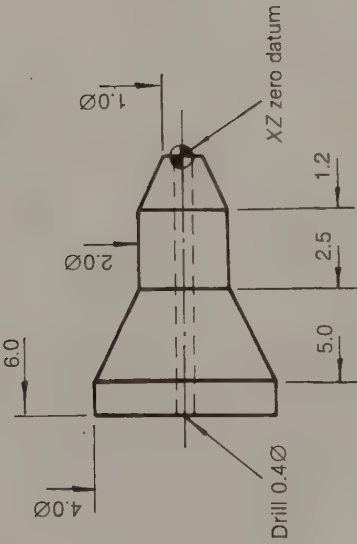
EXERCISE NO.	6
<i>Machining</i> Mill top face Mill 10 x 5 steps Mill pocket Drill six holes 5Ø, P1 to P6 <i>Tooling</i> Cemented carbide insert shell end mill, 60Ø, five teeth HSS end mill 10Ø HSS No. 2 center drill HSS drill 5Ø <i>Programming</i> Incremental dimensions Z datum clearance 2mm Linear interpolation Point-to-point positioning Cutter compensation Pocket mill cycle Quill clamp (steps and pocket) Drill cycle Spindle speed rev/min Feed mm/min Spray mist coolant Allocate tool offsets	The drawing shows a mechanical part with the following dimensions and features: Top View: Overall width 100, overall depth 72.50. A central rectangular feature has a width of 41.35 and a depth of 20. A fillet with a 5R radius transitions between the central feature and the outer edges. Six holes, labeled P1 through P6, are located on the top surface. Holes P1, P2, and P3 are in a row on the right side, with P1 at the outer edge and P3 30 units from the edge. Holes P4, P5, and P6 are in a row on the left side, with P4 at the outer edge and P6 30 units from the edge. The distance between the centerlines of the P1-P3 row and the P4-P6 row is 41.35. The distance from the centerline of the P1-P3 row to the right edge is 15. The distance from the centerline of the P4-P6 row to the left edge is 15. Side View: Overall length 80. The part has a stepped profile. The top surface is 10 units wide. The main body has a width of 59.98. The bottom surface is 8 units wide. The distance from the left edge to the start of the main body is 15. The distance from the end of the main body to the right edge is 5. Material: aluminum alloy (Metric Programming Exercise)

EXERCISE NO.	7
<i>Machining</i> Mill top face Mill angular steps Mill 90° slots <i>Tooling</i> Cemented carbide insert shell end mill, 5.0Ø, five teeth HSS end mill .625Ø <i>Programming</i> Incremental dimensions* Z datum clearance 2 inches Linear interpolation Quill clamp (steps and slots) Mirror image Spray mist coolant Spindle speed rev/min Feed inches/rev Allocate tool offsets Do not use cutter radius compensation	Technical drawing of a mechanical part with dimensions. The part has a total width of 8.0 and a total height of 4.0. The top surface is flat. The front face has a 90-degree slot on the left, 2.0 wide and 3.0 deep. The right side has a 90-degree slot, 1.5 wide and 1.005 deep. The bottom surface has a 90-degree slot, 1.5 wide and 1.005 deep. The part is symmetrical about a vertical centerline. A datum is indicated at the top left corner. Material: brass (Inch Programming Exercise)

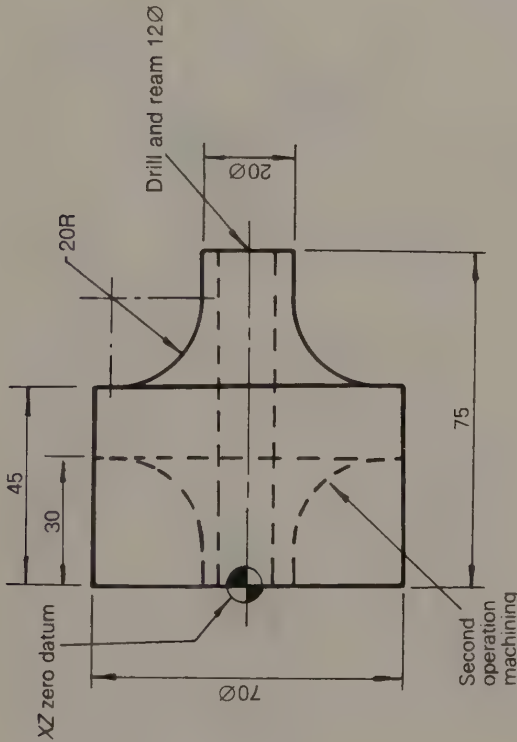
EXERCISE NO.	8
<i>Machining</i> Material supplied pre-machined 20 x 80 x 80 All other features to be machined Complete an operation schedule (see Figure 8.16). Detail tooling to be used Detail work-holding and setting arrangements Determine appropriate speeds and feeds Complete a part program Prove the part program Machine the component (Note: Z datum clearance 4mm)	 Material: aluminum alloy (Metric Programming Exercise)

EXERCISE NO.	9
Axes indicated assume a rear-mounted turret  Material: brass (Inch Programming) Exercise: See Example Program 9)	
<i>Machining</i>	Stock material 5.0Ø hand loaded and positioned Face one end Turn 4.5Ø, 3.5Ø and 2.5Ø, two passes per diameter Part off to length
<i>Tooling</i>	Light turning and facing, cemented carbide insert Parting off, cemented carbide insert .125 wide
<i>Programming</i>	Absolute positioning Linear interpolation Feed inches/rev Spindle speed rev/min Spray mist coolant Allocate tool offset numbers Ignore tool-tip radius

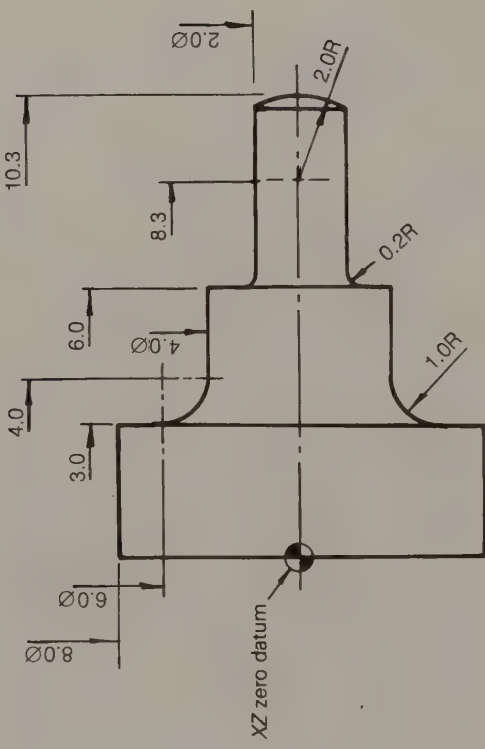
EXERCISE NO.	10
Axes indicated assume a rear-mounted turret  Material: mild steel (Metric Programming Exercise; see Example Program 10)	Machining Material pre-faced billet 100.Ø hand loaded and located against back face of chuck Face second end to length Center drill Drill 20Ø in two stages Bore 40Ø, 60Ø and 80Ø Depth of cut 2mm per pass Tooling No. 3 HSS center drill HSS drill 10Ø HSS drill 20Ø Internal turning tool, cemented carbide insert Facing tool, cemented carbide insert Programming Absolute dimensions Linear interpolation Feed mm/rev Surface cutting speed m/min Ignore tool-tip radius Allocate offset numbers Flood coolant

EXERCISE NO.	11
<i>Machining</i> Stock material 4.0$\varnothing$ hand loaded and positioned Face one end Center drill Drill 0.4$\varnothing$ Rough turn to profile Finish turn with continuous cut Part off <i>Tooling</i> Light turning and facing tool, cemented carbide insert HSS No. 2 center drill HSS drill .4$\varnothing$ Parting-off tool, cemented carbide insert 0.125 wide <i>Programming</i> Absolute dimensions Linear interpolation Feed inch/rev varied for roughing and finishing cuts Surface cutting speed feet/min varied for roughing and finishing cuts Flood coolant Peck drill cycle Ignore tool-tip radius Allocate tool offset numbers	 Drill 0.4$\varnothing$ XZ zero datum 6.0 4.0$\varnothing$ 5.0 2.5 1.2 2.0$\varnothing$ 1.0$\varnothing$ Material: brass (Inch Programming Exercise)

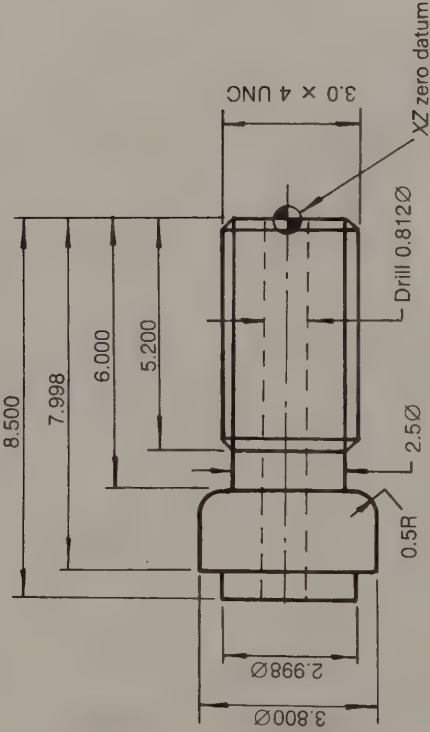
EXERCISE NO.	12
	<i>Machining</i> Material pre-faced billet 70Ø hand loaded and located against back face of chuck Face second end to length Center drill Drill in two stages Ream Rough turn to profile Finish turn profile <i>Tooling</i> Light turning and facing tool, cemented carbide insert, tip radius 2.0mm HSS No. 3 center drill HSS drill 6Ø HSS drill 11.5Ø HSS reamer 12Ø <i>Programming</i> Absolute dimensions Linear interpolation Circular interpolation Cutter radius compensation Feed mm/rev, varied for roughing and finishing cuts Surface cutting speed m/min, varied for roughing and finishing cuts Mist coolant Peck drill cycle for initial drilling Allocate tool offset numbers



Material: medium carbon steel
(Metric Programming
Exercise)

EXERCISE NO.	13
Machining Material pre-faced billet 8.0Ø hand loaded and located against back face of chuck Face second end to length Rough turn 4.0Ø and 2.0Ø Finish turn complete profile in one pass Tooling Roughing tool, cemented carbide insert, tip radius 0.200 Light turning and facing tool, cemented carbide insert, tip radius 0.100 Programming Absolute dimensions Linear interpolation Circular interpolation Cutter radius compensation Feed inches/rev varied for roughing and finishing cuts Surface cutting speed feet/min varied for roughing and finishing Mist coolant Allocate tool offset numbers	Material: mild steel (Inch Programming Exercise)  The technical drawing shows a stepped shaft with the following dimensions and features: Overall Length: 10.3 Step 1 (Left): Diameter 6.0Ø, length 3.0. A hole is located 4.0 from the left end. Step 2 (Middle): Diameter 4.0Ø, length 6.0. A fillet with radius 1.0Ø transitions to the next step. Step 3 (Right): Diameter 2.0Ø, length 8.3. A fillet with radius 0.2R transitions to the final step. Final Step (Right): Diameter 2.0Ø, length 2.0Ø. A fillet with radius 2.0R transitions to the end. Datum: XZ zero datum is located at the center of the 6.0Ø section.

EXERCISE NO.	14
Machining Material pre-faced billet 100Ø, hand loaded and located against back face of chuck Face second end to length Center drill Drill 18Ø in two stages Machine end radius in three passes Rough machine to profile Finish machine profile with continuous cut Tooling Light turning and facing tool, cemented carbide insert, tip radius 0.8mm Roughing tool, cemented carbide insert, tip radius 2mm HSS No. 4 center drill HSS drill 8Ø HSS drill 18Ø Internal turning tool, tip radius 1mm Programming Absolute dimensions Linear interpolation Circular interpolation Cutter radius compensation Feed mm/rev varied for roughing and finishing cuts Surface cutting speed m/min varied for roughing and finishing cuts Flood coolant Allocate tool offset numbers	Material: brass (Metric Programming Exercise)

EXERCISE NO.	15
<i>Machining</i> Stock material 4.0Ø, hand loaded and positioned Face end Turn 3.8Ø, 3.0Ø, 2.5Ø Cut screw thread 3.0 x 4 UNC Center drill Drill 0.812Ø Part off to length <i>Tooling</i> Light turning and facing tool, cemented carbide insert, tip radius .031 Grooving tool, cemented carbide insert, 0.250 wide Parting-off tool, cemented carbide insert, 0.125 wide Screw cutting tool, cemented carbide insert, unified national form HSS No. 3, center drill HSS 0.812 drill <i>Programming</i> Absolute positioning Linear interpolation Circular interpolation Cutter radius compensation Screw cutting cycle Peck drilling cycle Feed inches/rev Surface cutting speed feet/min No coolant Allocate tool offsets	 Technical drawing of a mechanical part. The part is a cylindrical component with a total length of 8.500. It features a central section with a diameter of 3.0Ø and a length of 6.000. The left end has a diameter of 3.800Ø and a length of 2.998Ø. The right end has a diameter of 2.5Ø and a length of 5.200. A center drill is located at the right end, with a diameter of 0.812Ø. The part is made of brass and is hand loaded and positioned. The drawing includes a datum line for the XZ zero point. Material: brass (Inch Programming Exercise)

EXERCISE NO.	16
<i>Machining</i> Material pre-faced billet 80Ø Machine to drawing Complete an operation schedule Detail tooling to be used Detail work-holding and setting arrangements Determine appropriate speeds and feeds Complete a part program Prove the part program Machine the component	The drawing shows a mechanical component with the following dimensions and features: - Overall length: 110 - Overall diameter: 80Ø - Section 1 (left): Length 35, diameter 70Ø. - Section 2 (middle): Length 40, diameter 50Ø. - Section 3 (right): Length 74, diameter 40Ø. - Thread: M65 x 1.5 - Other dimensions: 60, 50, 90, 105, 40Ø, 30Ø, 50Ø, 40Ø. - Datum: XZ zero datum is indicated at the center of the 70Ø section. Material: medium carbon steel (Metric Programming Exercise)

Assigned miscellaneous functions:

M00

M plus two digits

Programmed stop. Stops all slide movement, spindle rotation and coolant

M01

Optional stop. Ignored unless activated manually from control console.

M02

End of program

M03

Spindle on clockwise

M04

Spindle on counter-clockwise

M05

Spindle off

M07

Coolant on (mist)

M08

Coolant on (flood)

M09

Coolant off

M10

Quill clamp on

M11

Quill clamp off

M30

End of program. Rewind tape.

Axis movement commands:

End point in X, Y and Z

Start point of arcs relative to circle center I, J and K

Dimensional values in 3/2 format metric, that is, three digits before and two digits after the decimal point; 2/3 format inch. Do not program the decimal point and omit leading zeros.

Plus (+) signs are not required but (-) signs must be included.

Other functions:

Feed

F plus four digits

Feed/min in 1 mm or 0.1 in. steps

Feed/rev in 0.001 mm or 0.001 in. steps

Do not program decimal point or leading zeros

Spindle speed

S plus four digits

Tool identification

T plus two digits

T01 to T16. With offsets use four digits, first two tool number, second two offset number, offsets 00 to 32.

Dwell

D plus three digits in 0.1 s (Do not program decimal point.)

Slash delete. Messages ignored when 'slash delete' switch on the control unit console is activated. They are obeyed when switch is off.

%

Rewind stop

*

End of block

PROGRAMMING NOTES RELATING TO MILLING AND DRILLING EXERCISES

Pocket Clearance Cycle (G28)

1. Position the cutter over the center point of the pocket.
2. Program the appropriate movement in the Z axis.
3. Program G28 with X and Y values indicating pocket dimensions. Cutter radius compensation and the step-over value will be automatically determined and implemented.
4. Cancel the cycle by programming G80.

(Note this code is not available or standard on all machines.)

Mirror Image (G31)

1. Program G31 together with the axis or axes to be mirrored. For example, G31 X will mirror in the X axis only; G31 XY will mirror in the X and Y axes. No other data are to be included in this block.
2. Program the original axes commands.
3. Cancel by programming G30.

(Note this is not a standard code that is available on all machines, some machines will use a manual switch.)

Cutter Radius Compensation (G41, G42)

1. Program G41 or G42 in the same block as G01 or G00 when making the approach move to the profile to be machined. The cutter will offset by the appropriate radius.
2. Cancel G41 or G42 by programming G40 in the withdrawal move.

(Note some machines will require this code to be programmed in a separate block before the move.)

Drill Cycle (G81)

1. Program G81 when making the approach move to the first hole position. The block must also contain the Z depth to be drilled and Z clearance position. The drill cycle will activate at the end of the positional move and will be repeated at the end of each subsequent positional move until cancelled.
2. Cancel G81 by programming G80 in the withdrawal move.

Peck Drill Cycle (G83)

1. Program G83 when making the approach move to the first hole position. The block must also contain the Z depth to be drilled, the Z clearance position and the peck distance as a W or K increment. The drill cycle will activate at the end of the positional move and will be repeated at the end of each subsequent positional move until cancelled.
2. Cancel G83 by programming G80 in the withdrawal move.

ALTERNATIVE MACHINE SPECIFICATION AND CONTROL SYSTEM FOR PROGRAMMING EXERCISES 1 TO 8: MILLING AND DRILLING

Machine Type and Specification

Vertical machining center with three-axis control

Traverse:	X longitudinal 600 mm (24 in.) Y transverse 400 mm (16 in.) Z vertical 450 mm (18 in.)
Spindle speed:	10–3300 rev/min infinitely variable
Working surface:	1000 mm × 300 mm (40 in. × 12 in.)
Feeds:	1–5000 mm/min (0.1–198 in/min)
Tool changing:	Manual

Control System

Program format:	Word address
Axes controlled:	X, Y and Z singly or simultaneously
Interpolation:	Linear X, Y and Z Circular XY, YZ and ZX planes
Command system:	Incremental or absolute
Data input:	MDI or perforated eight-track tape
Tape code:	ISO or EIA via tape sensing

Programming Information

Block numbers:	N plus one to three digits
Preparatory functions:	G plus two digits
G00	Rapid traverse at machine maximum (modal)
G01	Linear interpolation, programmed feed (modal)
G02	Clockwise circular interpolation (modal)
G03	Counter-clockwise circular interpolation (modal)
G08	Ramp down inhibit (modal)
G09	Cancels G08
G28	Pocket clearance cycle (modal)

G30	Cancels mirror image
G31	Mirror image with axis command (modal)
G40	Cutter offset cancel
G41	Cutter radius compensation left (modal)
G42	Cutter radius compensation right (modal)
G70	Inch programming (modal)
G71	Metric programming (modal)
G80	Cancels all fixed cycles
G81	Drill cycle (modal)
G83	Peck drill cycle (modal)
G90	Absolute programming (modal)
G91	Incremental programming (modal)
G93	Inverse time feedrate (V/D) (modal)
G94	Feed inch (mm)/min (modal)
G95	Feed inch (mm)/rev (modal)
G96	Constant surface speed
G97	Spindle speed rev/min (modal)

ALTERNATIVE MACHINE SPECIFICATION AND CONTROL SYSTEM FOR PROGRAMMING EXERCISES 9 TO 16: TURNING

Machine Type and Specification

Precision turning center with two-axis control

Traverse:	X transverse 160 mm (6.3 in.) Z longitudinal 450 mm (17.7 in.)
Spindle speed:	50–3800 rev/min
Feed:	0–400 mm/min, 0–200 inch/min
Tooling:	Indexable turret providing eight tool stations

Control System

Program format:	Word address
Axes controlled:	X and Z singly or simultaneously
Interpolation:	Linear X and Z axes Circular XZ plane
Command system:	Absolute
Data input:	MDI and magnetic tape

Programming Information

Block numbers:	N plus one to three digits
Preparatory functions:	G plus two digits
G00	Rapid traverse at machine maximum (modal)
G01	Linear interpolation, programmed feed (modal)

G02	Clockwise circular interpolation (modal)
G03	Counter-clockwise circular interpolation (modal)
G08	Ramp down inhibit (modal)
G09	Cancels G08
G33	Threading cycle
G40	Cutter offset cancel
G41	Cutter radius compensation left (modal)
G42	Cutter radius compensation right (modal)
G70	Inch programming (modal)
G71	Metric programming (modal)
G80	Cancels all fixed cycles
G81	Drill cycle (modal)
G83	Peck drill cycle (modal)
G90	Absolute programming (modal)
G91	Incremental programming (modal)
G92	Preset axes command
G93	Feed inverse time (V/D)
G94	Feed inch or mm/min (modal)
G95	Feed inch or mm/rev (modal)
G96	Surface cutting speed feet or meters/min
G97	Surface speed rev/min
Assigned miscellaneous functions:	M plus two digits
M00	Programmed stop. Stops all slide movement, spindle rotation and coolant.
M01	Optional stop. Ignored unless activated manually from control console.
M02	End of program
M03	Clockwise spindle rotation
M04	Counter-clockwise spindle rotation
M05	Spindle off
M06	Tool change
M07	Coolant on (mist)
M08	Coolant on (flood)
M09	Coolant off
M30	End of program. Tape rewind.
Axis movement commands:	End point in X and Z
	Start point of arcs relative to circle center/and K (see pp. 128–136)
	Dimensional values in 3/2 format metric, that is, three figures before and two after the decimal point; 2/4 format inch. Program the decimal point but no leading or trailing zeros.
	Plus (+) signs are not required but (–) signs must be included.
	X values to be programmed as a diameter.
Other functions:	
Feed	F plus four digits
	Feed/min in 1 mm or 0.1 in. steps
	Feed/rev in 0.001 mm or 0.001 in. steps
	Do not program decimal point
Spindle speed	S plus four digits

Surface speed	S plus three digits
Tool identity	T plus two digits, T01 to T10. With offsets program four digits, offsets 00 to 24, first two digits tool number/second two offset number.
%Rewind stop	Slash delete. Messages ignored when 'slash delete' switch on the control unit console is activated. They are obeyed when switch is off.
*End of block	

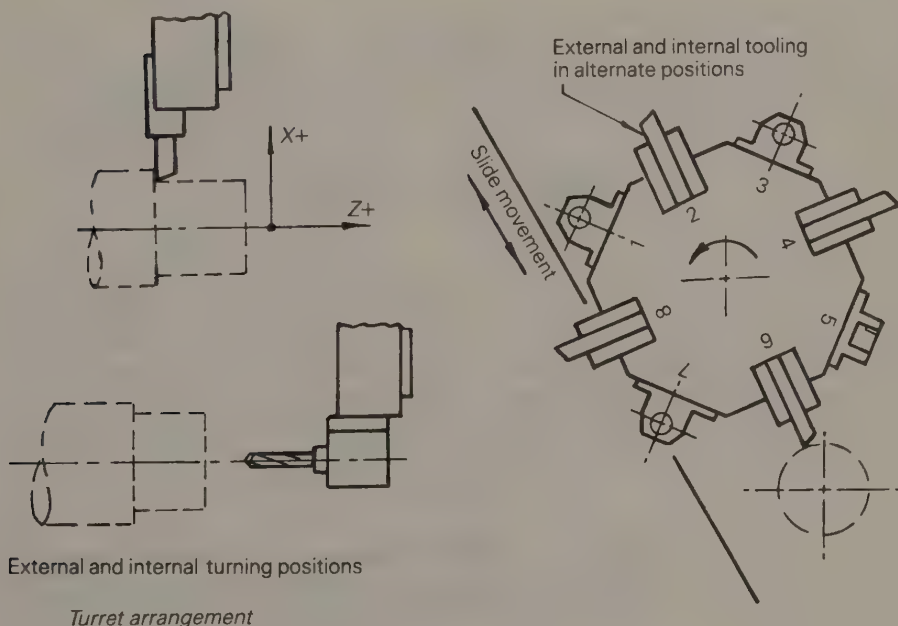
PROGRAMMING NOTES RELATING TO TURNING EXERCISES

Tooling System and Turret Indexing

The tooling system comprises an eight-station turret mounted at 90° to the machine spindle axis, the tooling positions being numbered 1 to 8. (See following figure.) It illustrates the way the tooling is arranged in relation to the work-piece.

Turret indexing is achieved as follows:

1. In the block following the end of a machining sequence program M06 (tool change). No other data are to be included in this block. The turret will withdraw to a pre-set safe indexing position.
2. In the next block program the required tool number and, if applicable, its related tool offset number. The turret will index by the shortest route to that tool position.



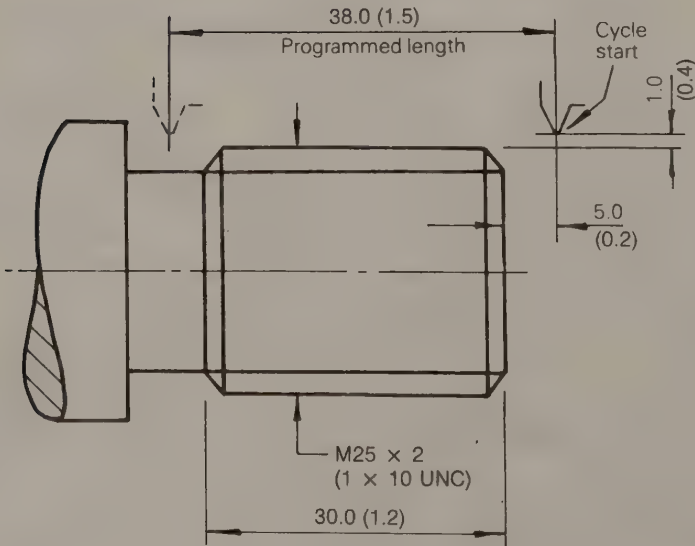
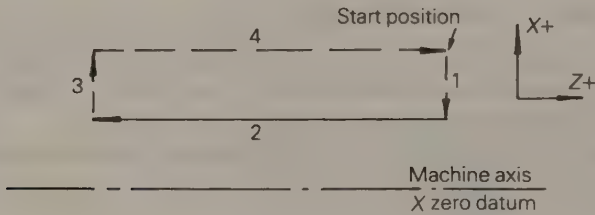
- 3. In the next block program a rapid move to the next pre-cutting position, which should be approximately 2mm or 0.100 in. clear of the work-piece.
- 4. Finally, continue to program moves at a controlled feedrate.

Note tool changes for various machines will program differently and machinery manuals should be consulted.

Screw-cutting Cycle (G33)

The movement sequence of the screw-cutting cycle is illustrated in the following figure.

- 1. Tool moves rapidly to programmed X value.
- 2. Thread is cut to programmed Z value.
- 3. Rapid traverse to initial position in X axis.
- 4. Rapid traverse to initial position in Z axis.



Component detail. (Inch units are given in parentheses.)

The final thread depth is reached by a series of rough and finish pass cuts as indicated in the part program below, which refers to the preceding figure.

The pitch is programmed as a *K* value.

Allow an approach distance of 0.250 inches in the *Z* axis when cutting threads.

(Inch)	<i>N</i>	<i>G</i>	<i>X</i>	<i>Z</i>	<i>I</i>	<i>K</i>	<i>F</i>	<i>S</i>	<i>T</i>	<i>M</i>	*
35		0	2.7	.25							*
40		33	2.48	-3.8		2		490	0404		*
45			2.46								*
50			2.44								*
55			2.42								*
60			2.4								*
65			2.38								*
70			2.36								*

Allow an approach distance of 5 mm in the *Z* axis when cutting threads.

(Metric)	<i>N</i>	<i>G</i>	<i>X</i>	<i>Z</i>	<i>I</i>	<i>K</i>	<i>F</i>	<i>S</i>	<i>T</i>	<i>M</i>	*
35		0	26	5							*
40		33	24.8	-38		2		490	0404		*
45			24.6								*
50			24.4								*
55			24.2								*
60			24								*
65			23.8								*
70			23.6								*

The *X* values are continually reduced until the full thread depth is reached. Two passes of the final cut should be made to reduce tool pressure for finishing.

The *Z* axis move is programmed as an incremental value from the start position.

The cycle is cancelled by programming G80.

Drill Cycles (G81 and G83)

1. Position drill at start point in the *Z* axis on *X* zero datum.
2. Program G81 or G83 together with *Z* depth to be drilled. G83 requires the peck distance to be programmed as a special coded increment. (Refer to machine manual for address used.)
3. Cancel G81 or G83 by programming G80.

Cutter Radius Compensation (G41 and G42)

Common to both turning and milling exercises.

Exercise 10 (Metric Programming Example)

PART PROGRAM EXERCISE NO. 10				MACHINE As book				CONTROL As book				
				COMPILED BY A. Programmer				DATE 5-6-84				
N	G	X	Y	Z	I	J	K	F	S	T	M	REMARKS
01								%				* Rewind stop
05	71											* Metric
10	95											* Feed mm/rev.
15	96											* Speed m/min.
20										0202		* Tool and offset
25	00	104		0								* Rapid to start
30											03	* Spindle on
	01	0		2				250	170		08	* Feed to X0, coolant on
35	00											* Clear face
40											06	* Tool change
45										0101		* Index to c/drill
50		0		2								* Rapid to start
55	01			-5				80	28			* Center drill
60	00			2								* Return to start
65											06	* Tool change
70										0303		* Index to drill 10φ
75		0		2								* Rapid to start
80	83			-85				W 20	180			* Peck drill 10φ
85	80											* Cancel cycle
90											06	* Tool change
95										0505		* Index to drill 20φ
100	00	0		2								* Rapid to start
101	81			-85				300				* Drill 20φ

APPENDIX D

GDT SYMBOLS

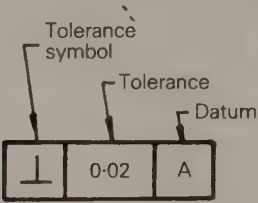
Geometric tolerances

Geometrical tolerance symbols

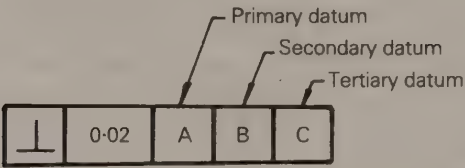
SYMBOL	CHARACTERISTIC	APPLICATION
	STRAIGHTNESS	Applied to an edge line or axis. For an edge or line the tolerance zone is the area between two parallel straight lines containing the edge or line. The tolerance value is the distance between the two lines.
	FLATNESS	Applied to a surface. The tolerance zone is the space between two parallel planes. The tolerance value is the distance between the two planes.
	ROUNDNESS	Applied to the cross-section of a cylinder, cone or sphere. The tolerance zone is the angular space between two concentric circles lying in the same plane. The tolerance value is the distance between the two circles.
	CYLINDRICITY	Applied to the surface of a cylinder. Combines roundness, straightness and parallelism. The tolerance zone is the angular space between two coaxial cylinders. The tolerance zone is the radial distance between the two cylinders.
	PROFILE OF A LINE	Applied to a profile. The tolerance zone is an area defined by two lines that have a constant width normal to the stated profile. The tolerance is the diameter of a series of circles contained between the two lines. The tolerance may be unilateral or bilateral.
	PROFILE OF A SURFACE	Applied to a surface. The tolerance zone is a space contained between two surfaces normal to the stated surface. The tolerance value is the diameter of a series of spheres enveloped by the two surfaces. The tolerance may be unilateral or bilateral.
	PARALLELISM	Applied to a line, surface or cylinder. The tolerance zone is the area between two parallel lines or planes, or the space between two parallel cylinders, which must be parallel to the datum feature. The tolerance is the distance between the two lines or planes or, in the case of a cylinder, the diameter of the cylinder.
	SQUARENESS	Applied to a line, surface or cylinder. For a line or surface the tolerance zone is the area between two parallel lines or planes which are perpendicular to the datum surface. The tolerance is the distance between the lines or planes. For a cylinder the tolerance zone is the space within a cylinder equal in diameter to the tolerance value and perpendicular to the datum plane.
	ANGULARITY	Applied to a line, surface or cylinder. For a line or surface the tolerance is the area or space between two parallel lines or planes inclined at a specified angle to the datum feature. For a cylinder the tolerance zone is the space within a cylinder equal in diameter to the tolerance value and inclined at a specified angle to the datum feature.
	POSITION	Applied to a circle or cylinder. The tolerance zone is the space within a cylinder equal in diameter to the tolerance value and coaxial with the datum axis. The tolerance limits the deviation of the datum axis from its true position.
	CONCENTRICITY	Applied to parallel lines or surfaces. The tolerance zone is the area of space between the lines or surfaces symmetrically disposed in relation to a datum feature. The tolerance value is the distance between the lines or planes.
	SYMMETRY	Applied to a point, axis, line or plane. The tolerance zone definition varies according to the feature. The tolerance value will limit the positional deviation from the specified true position.
	RUN OUT	Applied to the surface of a solid of revolution or to a face perpendicular to the axis. The tolerance value indicates the permissible indicator movement during one revolution.
	MAXIMUM MATERIAL CONDITION	MMC exists when the component or feature contains the maximum amount of material permitted by its dimensional tolerances. When M is included in a tolerance frame the tolerance value need only be applied rigorously when the component or feature is in that condition. When not in that condition the geometric tolerance may be increased up to the difference between the MMC limit and the actual finished size.

Geometric tolerance frames

1. Tolerance relating to a single datum:



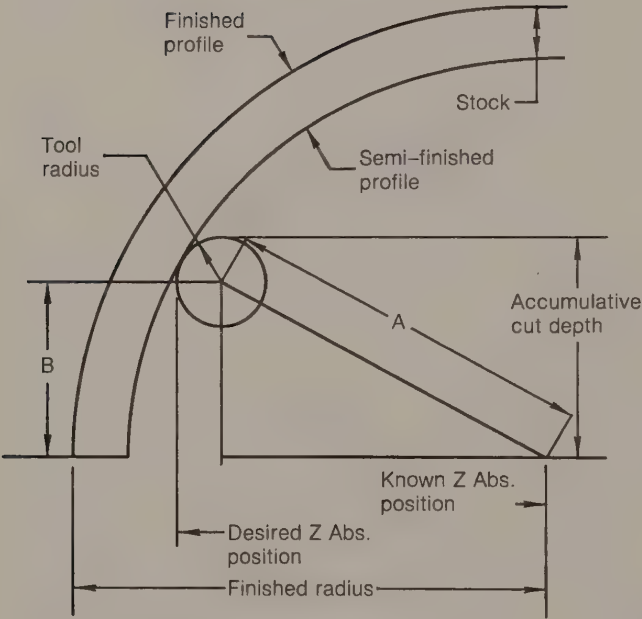
2. Tolerance relating to more than one datum:



APPENDIX E

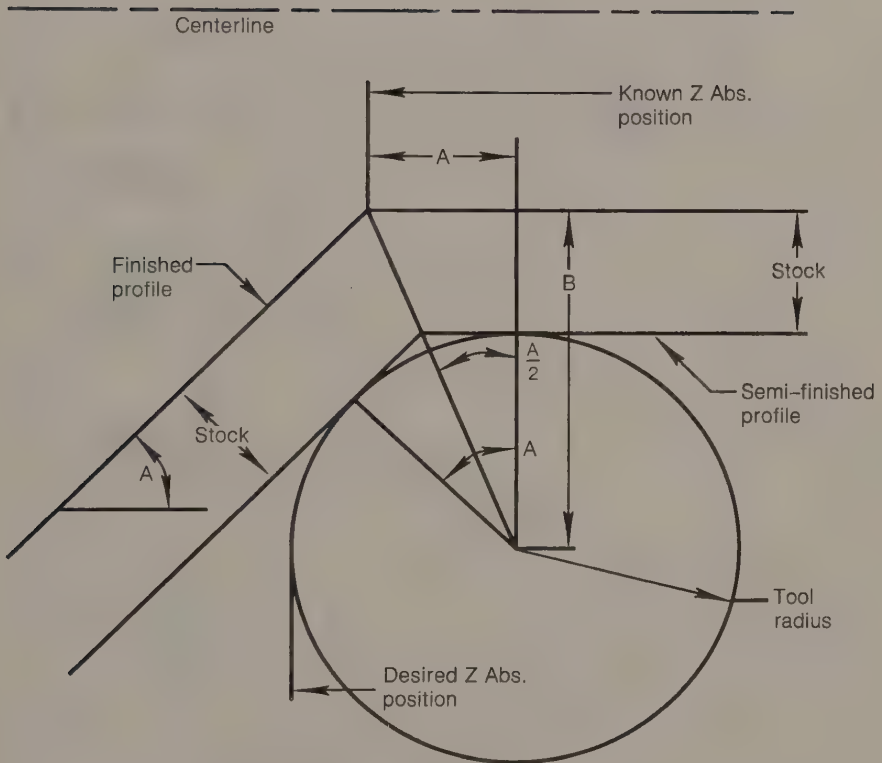
FORMULAS

FORMULA #1



A = Finished radius-stock-tool radius
B = Accumulative cut depth-tool radius
Desired Z Abs. = Known Z Abs. - $\sqrt{A^2 - B^2}$ - tool radius

FORMULA #2

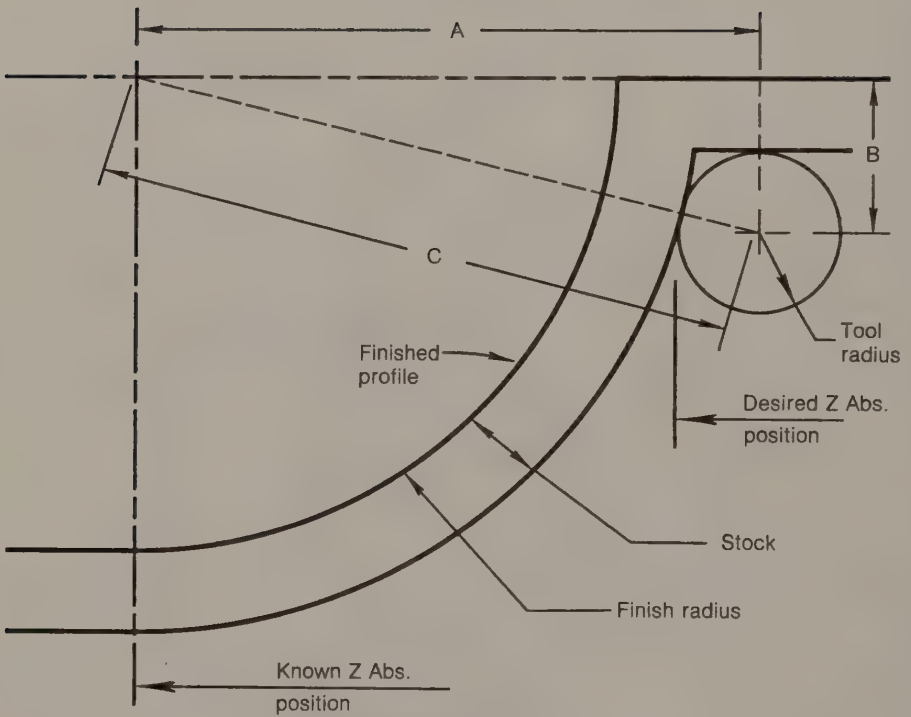


$$B = \text{Stock} + \text{tool radius}$$

$$A = \tan \frac{A}{2} \times B$$

$$\text{Desired Z Abs.} = \text{Known Z Abs.} - (\text{tool radius} - A)$$

FORMULA #3

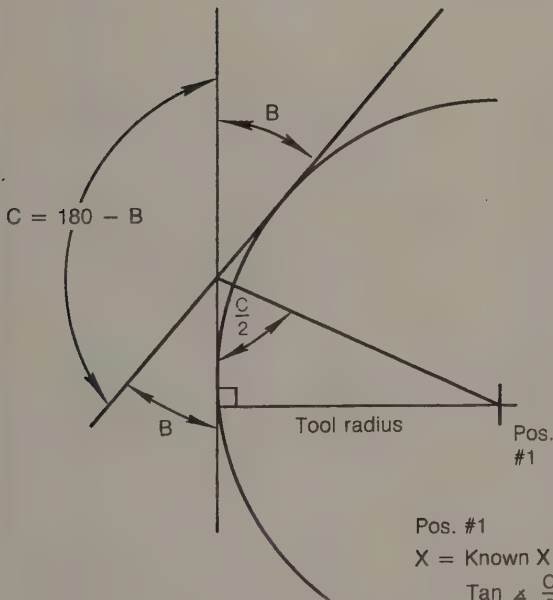
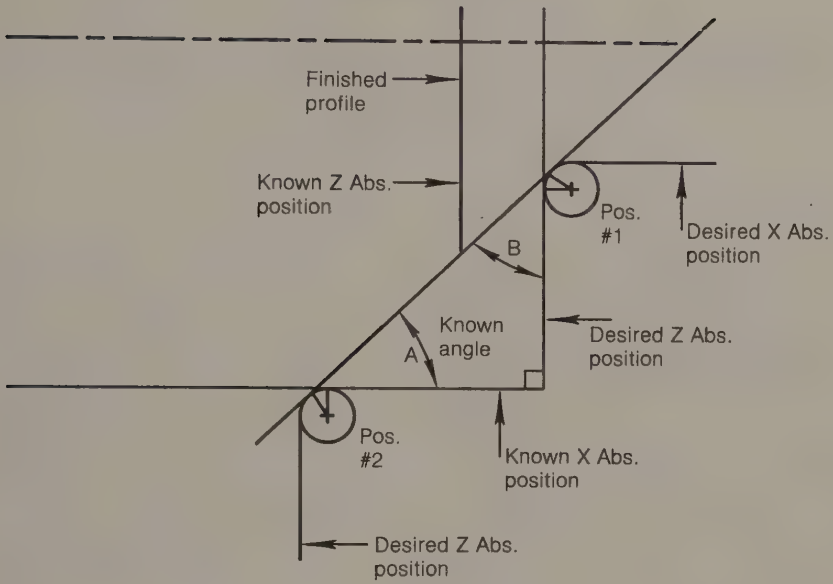


$$B = \text{Tool radius} + \text{stock}$$

$$C = \text{Finish radius} + \text{stock} + \text{tool radius}$$

$$A = \sqrt{C^2 - B^2}$$

$$\text{Desired Z Abs.} = \text{Known Z Abs.} + A - \text{tool radius}$$

FORMULA #4

Pos. #1

$$X = \text{Known } X - \tan \angle A (\text{chamfer length} + \text{clearance}) + \tan \angle \frac{C}{2} (\text{tool radius}) - \text{tool radius}$$

$$Z = \text{Known } Z + \text{clearance}$$

Pos. #2

$$X = \text{Known part radius}$$

$$Z = \text{Known } Z - \text{chamfer length} + \tan \angle \frac{C}{2} (\text{tool radius}) - \text{tool radius}$$

INDEX

- absolute positioning, 125–128, 182, 199
- acceleration, 136, 137
- accounts payable, 458
- Acme, 23, 24
- ADAPT, 371
- adaptive control, 444–447
- addresses, 193
- alternating current (AC) drive, 22, 23
- American National Standards Institute (ANSI), 43, 44, 137
- American Standard Code for Information Interchange (ASCII), 104
- APT, 371
- arc center, 185, 205, 234
- area clearance, 240
- automatic guided vehicle (AGV), 442
- automatic tool change, 59, 176
- Automatically Programmed Tools (APT), 345, 371–379
 - geometry statements, 373, 375, 376
 - initialization statements, 373, 375
 - tool change statements, 373, 377
 - tool motion statements, 373, 377–379
 - termination statement, 373, 379
- AUTOSPOT, 371
- axes of motion, 7, 13, 195–198
- axis identification, 193
- backlash, 23
- ballscrew, 23–25
- bar code, 442
- bar feeder, 92, 93
- BASE statement, 357
- batch, 364
- bench mark, 335
- binary code, 101–103, 105
- billet, 93
- bit, 105
- block, 105, 182, 193, 284
- block-by-block run, 109, 245
- block delete, 145–147, 232, 233
- block numbers, 183, 184, 193
- bolt hole circle, 141–143, 290, 291
- brazed tip, 42
- buffer storage, 114
- business planning, 453
- calculations, 233–238, 309–333
 - exercises, 314–333
- caliper, 447
- canned cycle, 99, 122, 137, 138, 213–217, 334
- capacity requirements planning, 458
- carbide grades, 41, 43
- cartridges, 43, 50, 56–58
- cassettes, 112
- cast iron structures, 20
- cathode ray tube (CRT), 97, 341
- cemented carbide, 41, 42
- center locator, 168
- center point, 202
- central processing unit (CPU), 391
- ceramic structures, 20
- character, 105
- check surface, 377
- chip breaker, 42, 43
- CIMCAM, 401, 402–433
- Cimlinc, 394–433
- circular interpolation, 86, 122, 124, 128–136, 194, 199–213, 334
 - multiquadrant, 135, 202–211
 - single block, 211
 - 360°, 135, 205, 211–213
- CL data, 346, 362–364, 379, 381–386
 - Compact II, 346, 362–364, 379
 - APT, 381–386
- clockwise movement, 7, 13, 128, 202, 203
- closed loop, 11, 29, 445

- CNC program, 427
- column, 103
- Compact II, 345, 352, 354–370, 371
 - examples, 365–370
 - geometry statements, 357–358
 - initialization statements, 355–357
 - tool change statements, 358–360
 - tool motion statements, 360–362
 - termination statement, 362
- compensation, 67, 71–74, 225–232, 295, 449
- computer-aided/assisted part
 - programming, 335, 344–386
 - APT, 371–379
 - Compact II, 352–371
 - language based, 346–386
- computer-aided design/computer-aided manufacturing (CAD/CAM), 336, 371, 389–391, 449, 450
- computer-aided engineering (CAE), 390
- computer-aided process planning (CAPP), 335
- computer installation, 335
- computer-integrated manufacturing (CIM), 339, 391, 449–459
- computer numerical control (CNC), 1, 3
 - capability, 3
- computer program, 427
- concrete structures, 20
- conditional expressions, 224
- constant surface cutting speed, 154
- continuous loop tape, 100
- continuous path machining, 122, 123
- contouring, 122, 123, 194
- conversational data input, 97, 116, 119, 162
- conversational part programming, 284–308
 - examples, 296–306
- conveyor, 442
- coordinate dimension, 9
- counterclockwise movement, 7, 14, 128, 202, 203
- CRT screen, 97
- cursor, 342, 394
 - graphics, 394
 - icon, 394
 - popup, 394
- cutter diameter compensation/offset, 229–231
- cutter image, 388
- cutter location data file, 346, 362–364, 379, 381–386
 - APT, 381–386
 - Compact II, 346, 362–364, 379
- cutter radius compensation/offset, 229–231, 387, 388
- cutting speed, 97, 152, 193
- cutting tools, 41, 358
- data format, 191–194
- data input, 9, 95, 97
 - conversational, 97
 - manual, 97
- data preparation, 95, 96
- datums, 80–82, 85, 148, 149, 167, 198, 233
 - fixed machine, 148
 - machine, 148, 167, 198
 - program, 148, 149, 167, 198
 - zero, 148, 167
- deceleration, 136, 137
- deflection, 21
- diameter compensation/offset, 229–231
- digitizing, 439
- digitizing tablet, 341, 391
- dimensional accuracy, 80–82
- dimensional stability, 33–36
- direct current (DC) drive, 22, 23, 28
- direct numerical control (DNC), 114, 336, 339, 440
- documentation, 173–182
 - operation sheet, 173, 187, 188
 - program listing, 187
 - setup sheet, 173–175
 - tool sheet, 179, 188
- drive surface, 377
- dry run, 109, 243
- dynamic loading, 17–19
- Electrical Industries Association (EIA), 104, 105, 116–120, 124, 137, 183
- electronic probe, 168
- electronic storage unit, 114
- elliptical profile, 334
- emergency stop, 233
- encoder, 30, 31
- end effector, 93
- end point, 202
- fabricated steel structures, 20
- face milling cycle, 140, 289

- family of parts, 222, 436
- fanfold tapes, 100
- feed, 7, 152, 157–160, 182, 184, 193, 358
- feedback, 11, 445
- feedrate, 97, 157–160, 184, 193, 285
- feedrate override, 159, 160
- fiber optic tape reader, 99, 102
- financial analysis, 458
- financial management and business control, 458
- finish point, 131–126, 205
- fixed block format, 105, 191, 192
- fixed cycles, 137–145, 213
 - nonstandardized, 139–145, 213
 - standardized, 137, 138, 213
- fixed machine datum, 148
- fixed sequence, 191, 192
- fixed zero, 198
- flexible manufacturing system (FMS), 439–444
- floating zero, 148, 198
- floppy disk, 114
- format, 105, 191, 192
 - fixed block, 105, 191, 192
 - tab sequential, 105
 - tape, 105
 - variable block, 105, 191, 192
 - word address, 105, 191
- form tool, 176
- gage length programming, 226
- G code, 116–118, 124, 183–185, 187, 193
- general ledger, 458
- geometric construction, 347, 348, 372
- geometric definition, 347–352, 357–358, 372, 375–376
 - circle, 348, 350, 353, 365, 373, 375, 376
 - direction, 347, 350
 - identity, 351, 352, 372
 - line, 347, 349, 353, 365, 373, 375, 376
 - point, 347, 348, 353, 365, 373, 375, 376
- geometric elements, 346, 372
- geometry labels, 357, 372
- geometric tolerances, 167
- geometry verifications, 357
- graphical numerical control (GNC), 386
- graphic image, 342
- graphics cursor, 394, 395
- graphics icon, 391
- graphics screen, 391
- gripper, 93
- gothic arch, 24
- grid plate, 84, 85, 87
- helical interpolation, 122, 292
- hardware configuration, 338
- high-speed steel, 41
- horsepower monitoring, 13
- hydraulic clamping, 14
- icon, 394, 404
- icon cursor, 394, 396
- IDENTification statement, 356
- inch programming, 154
- inches per minute, 7, 157
- inches per revolution, 7, 157
- inches per tooth, 157, 158
- incremental positioning, 125–128, 182, 199
- indexable inserts, 42, 43, 45, 47, 48, 52, 53, 176
- indexer, 13, 86–88, 198
- information flow, 451
- initialization statements, 355–357, 373, 375
 - APT initialization, 373, 375
 - BASE, 355–357
 - IDENT, 355–357
 - INIT, 355–357
 - MACHIN, 355–357
 - SETUP, 355–357
- INITialization statement, 356
- inprocess measurement, 11, 447–449
- input media, 97, 99, 112, 114, 341, 391
 - conversational input, 97
 - direct numerical control (DNC), 114
 - electronic storage unit input, 114
 - keyboard input, 341, 391
 - magnetic (floppy) disk input, 114
 - magnetic tape input, 112
 - manual data input (MDI), 97
 - tape input, 99
- integration, 459
- interactive, 346, 358, 364
- interactive headline, 391
- internal stops, 84
- International Standards Organization (ISO), 43, 46, 104, 105, 183
- I value, 131–136, 203, 204

- inventory management, 457
- inverse video, 394
- jog, 147
- jog retract, 147
- joystick, 342, 391
- J value, 131–136, 203, 204
- K value, 131–136, 203, 204
- laminates, 99
- language-based system, 346–386
- laser, 449
- leading zero, 193
- leadscrew, 23
- lead time, 451
- left mouse button (LMB), 392, 394, 395
- length offset/compensation, 69–71
- light pen, 341, 391
- limit switch, 442
- linear coordinates, 199
- linear interpolation, 119–122, 124, 194
- location, 78–86, 166, 170
- logon, 336
- loop programming, 137, 139, 220, 290
- machine control unit (MCU), 9
- machine datum, 148, 198
- machine language, 364
- machine layout, 355
- machine slide movement, 6, 13, 26, 27
- machine specifications, 163, 164
- machine spindle, 13, 21–23
- MACHINE statement, 356
- machine structure, 20
- machine zero, 149, 198
- machine center (vertical/horizontal), 17, 18, 124, 196, 197
- machining cell, 440–444
- machining curves, 397, 400
- machining mode, 406
- macro, 137, 144, 213, 221, 436
- magazine, 176
- magnetic disk (floppy), 114
- magnetic tape, 112, 113
- mainframe, 335, 336, 391
- man-readable message, 233
- manual data input (MDI), 97, 124, 162
- manual part programming, 162, 245–284
 - examples, 245–284
- MAPICS (Manufacturing, Accounting and Production Information Control System), 456
- marketing and physical distribution, 458
- master production schedule planning, 458
- material file, 285, 286
- materials requirements planning, 458
- mean value, 213
- menu, 341, 342
- middle mouse button (MMB), 392, 394, 395
- mini, 336, 391
- meters per minute, 7
- metric programming, 154
- M function/code, 119, 120, 183, 184, 187, 193
- microcomputer (micro), 334, 335
- millimeters per minute, 7, 157
- millimeters per revolution, 157
- millimeters per tooth, 157, 158
- minus (–), 105
- minus movement, 7, 124
- mirror image, 144–146, 225, 295
- miscellaneous function, 118–120, 183, 184, 187, 193
- modal, 116–118, 184
- monitoring, 442
- mouse, 342, 391, 392, 394–396
- mouse button, 392, 394, 395
 - left mouse button (LMB), 394, 395
 - middle mouse button (MMB), 394, 395
 - right mouse button (RMB), 394, 396
- multiaxis contour programming, 371
- Mylar, 99
- nesting, 143–145, 218
- network, 336
- nonstepping motor, 28
- numerical control (NC), 1
- offline, 335
- offsets, 64, 68–74, 225–232, 449
- open loop, 11, 28, 29, 448
- operation sheet, 173, 188
- operation execution, 453
- operation management, 453
- optical gratings, 31, 32
- optional stop, 148, 233
- order entry, 458

- overhang, 21
- override, 152, 159, 160
- pallet, 88, 89, 440
- parameter, 221, 436
- parametric programming, 144, 221–224, 436–439
- parametric variables, 144
- parity, 105
- part boundary, 358
- part geometry, 357
- part programming, 116, 162–308
- pattern, 137
- peck drill cycle, 214
- perforated tape, 99
- photocells, 31
- photoelectric device, 442
- photoelectric tape reader, 99, 102
- pitch circle, 141, 142
- plant operations, 457
- plotter, 109
- plus (+) movement, 7, 124
- pneumatic clamping, 14
- pocket, 240
- pocket milling cycle, 140, 141, 240, 290, 291
- point definition, 224, 225
- pointer strokes, 391
- point-to-point positioning, 119, 121, 194
- polar coordinates, 86, 135, 199
- popup cursor, 394, 395
- popup menus, 391, 393, 399
- positional feedback, 11, 29–32
- postprocess, 162, 345, 346, 364, 386, 387, 427
- preparatory function, 116–118, 183–187, 193
- preset tooling, 64, 66–68, 179
- preset zero, 167
- probe, 446, 449
- procedure, 163, 164
 - part programming procedure, 182
- process monitoring/tracking, 451
- processing, 165, 166, 309, 335
- processing center, 444
- production control, 457
- product data management, 458
- production monitoring and control, 458
- production planning, 458
- profile intersection point, 234
- program datum, 148, 149, 167, 198
- program definition list, 352
- program listing, 187, 189–191, 242–245
- program proving, 95, 242–245
- program stop, 148, 233
- program zero, 167, 357
- programming, 95, 162–308
- proving, 242–245
- proximity sensor, 442
- puck, 341, 391
- punched tape, 106–108
- purchasing, 458
- Pythagorean Theorem, 310, 311
- quadrant circular interpolation, 202, 204–213
- qualified tooling, 66, 179
- radius offset/compensation, 71–74, 229, 231, 387, 388
- ramp down, 136, 137
- ramp off/on, 231
- ramp up, 136, 137
- repeatability, 16
- replacement tooling, 179
- resolver, 31
- revolutions per minute, 154, 155
- rewind, 182
- rewind stop code, 182
- right mouse button (RMB), 394, 396
- robot work-handling devices, 339, 440
- rotary axes, 7, 13, 198
- rotary movement, 13, 198
- rotary table, 13, 86–88, 198
- rotate, 142, 143
- rotation, 202, 203, 292, 293
- routines, 99
- safety, 37–39, 45, 55
- scaling, 145, 147, 293–295
- screen menu, 342
- screw cutting cycle, 287, 289
- servo motor, 6
- set-up, 173–175, 355
- SETUP statement, 356
- sequence mode, 406
- sequence of operation, 162, 165
- simulation software, 444
- simultaneous engineering, 451
- sine rule, 312–314
- single cycle, 109
- single keyboard input, 391

- sister tool, 446, 447
- slash delete, 145–147, 232, 233
- slide drive, 28
- slide movement, 6, 9–11, 124, 182, 194–198
 - complex movement, 9
 - dimensional definition, 124
 - synchronizing movement, 10
 - verification of movement, 11
- slot milling cycle, 140, 141, 290
- soft jaws, 84
- sound sensor, 447
- source program, 354–370, 373–379
 - Compact II source, 354–370
 - APT source, 373–379
- span, 397
- special tool, 176
- specifications, 163, 164
- spindle, 13, 21–23
- spindle drive, 22, 23
- spindle speed, 97, 154–156, 184, 193, 285, 358
- spindle speed override, 159, 160
- start point, 131–136, 202
- static icon, 394, 404
- static load, 17–19
- static menu, 394
- stator, 30, 31
- stepping motor, 28
- steady rest, 90, 91
- stock removal, 240, 286
- storage retrieval system, 440
- stored cycle, 144
- stylus, 341, 446
- subroutine, 137, 144, 218
- super micro, 337
- super mini, 336
- supplier performance, 451
- surface cutting speed, 156
- surface finish, 153
- surface sensing probe, 446
- surface roughness file, 285, 286
- tab sequential format, 105
- tablet, 391
- tailstock, 89
- tape format, 105
- tape input, 99–112
- tape proving, 109, 243, 245
 - block-by-block run, 109, 245
 - dry run, 109, 243
 - single-cycle run, 109
 - test run, 109, 243
- tape reader, 99–102
 - fanfold tape, 100
 - fiber optic reader, 102
 - loop tape, 100
 - photoelectric reader, 102
 - reel tape, 101
- tape standards, 101, 104, 106
- target point, 131–136, 205
- telephone modem, 336
- temperature, 33–36, 445
- terminal, 336
- termination statement, 362, 373, 379
 - Compact II (END/FINISH), 362
 - APT (FINI), 373, 379
- test run, 109, 243
- text macro processor, 391
- three-dimensional CAD/CAM, 391
- thrust, 21
- time share, 336
- titles (popup menus), 394
- tolerance band, 213
- tool address, 193
- tool change data statement, 358–360, 373, 377
 - Compact II statement, 358–360
 - APT statement, 373, 377
- tool change position, 178
- tool changing, 14, 59
- tool clearance plane, 214
- tool compensation, 225–232, 295, 449
- tool holders, 43, 49, 51, 54, 55
- tool identity, 59, 176, 184
- tool layout, 181
- tool length offset, 168
- tool library, 360, 397
- tool life, 153
- tool magazines, 62–65, 176
- tool motion statements, 360–362, 373, 377–379
 - Compact II, 360–362
 - APT, 373, 377–379
- tool offset, 225–232
- tool path, 238–242, 363
- tool radius offset/compensation, 229–231, 449
- tool selection, 176
- tool sheet, 180, 188
- tool station, 59
- tool storage, 176
- tooling system, 59, 60
- torque, 23, 445, 446
- torque monitoring, 13, 445, 446
- torque variation, 445, 446

- total machine control, 5
- touch sensor, 449
- track, 103
- tracker ball, 342
- trailing zero, 193
- transducer (rotary, optical), 29, 30, 33, 449
- translate, 162, 292, 293
- trigonometrical ratios, 311–314
- tumble box, 101
- tungsten carbide, 41
- turning center, 17, 18, 124, 196, 197
- turrets, 61, 62, 176
- turret reference point, 355, 357
- two-dimensional CAD/CAM, 391
- UNIAPT, 371
- units, 182
- user-defined cycles, 144, 217–220
- user interface device, 391
- variable block format, 105, 191, 192
- vector refresh, 342
- vision, 449
- visual display unit, 178
- warning, 446
- wiggler, 168
- word, 105
- word address format, 105, 162, 182–185, 191
- work chutes, 444, 445
- work holding (mechanical, pneumatic and hydraulic), 14, 78, 79, 166
- work loading (manual and automatic), 91–93
- work location, 170
- work station, 336
- work support, 88–91
- zero, 148, 149, 167, 198, 357
 - absolute zero, 357
 - fixed zero, 198
 - floating zero, 148, 198
 - machine zero, 149, 167, 198
 - preset zero, 167
 - program zero, 149, 167, 198
 - reference zero, 167
 - zero datum, 148, 167
 - zero offset, 149, 150, 198
 - zero shift, 84, 148, 198
- zero datum, 148, 167
- zero gage length programming, 226
- zero offset, 149, 150, 198
- zero shift, 84, 148, 198

INDUSTRIAL PRESS INC.
200 Madison Avenue
New York, NY 10016-4078



90000

9 780831 130091

ISBN 0-8311-3009-1